

UNIVERSITAT DE LLEIDA
ESCOLA POLITÈCNICA SUPERIOR
ENGINYERIA TÈCNICA EN INFORMÀTICA DE SISTEMES

TREBALL FINAL DE CARRERA

Estudi i Disseny d'un Sistema de
Modificació Tímbrica en
Instruments Determinats a
través d'Anàlisi, Filtratge i
Síntesi de Senyal d'Àudio Digital

Autor: Marcel Claramunt Hernando

Director: Francisco Clariá Sancho

Octubre 2014

Motivació Personal

La música és per mi molt més que un hobby, és part de la meva educació, una ciència que incideix en els sentits i les emocions, i una forma d'expressió impressionant.

Des de sempre hi he estat molt vinculat, començant amb les classes de solfeig i continuant amb el clarinet com a primer instrument, em vaig encuriosir amb altres estils musicals diferents al de la música clàssica. Va ser amb la música electrònica com vaig descobrir la producció musical des de l'ordinador, i tot l'ampli ventall d'eines per a mesclar i crear sons, compondre peces musicals, programar controladors,... eines que fan que un simple ordinador pugui esdevenir un complet estudi de gravació o encara més, un laboratori de so.

Aquí és on entren les noves tecnologies, quelcom que també m'apassiona. I és per això que durant el batxillerat vaig fer un treball de recerca sobre un controlador MIDI que es podia utilitzar com a disparador de sons, convertint-se en una bateria electrònica casolana.

Un cop a la universitat, i finalitzada l'assignatura de TRACSE, amb nous coneixements adquirits com ara les funcions de transferència, els filtres, Fourier, entre d'altres, i sobretot l'eina Matlab de les sessions pràctiques, va ser quan vaig intentar buscar un projecte TFC que pogués néixer fruit d'aquests coneixements, aplicats a un àmbit del meu interès i buscant quelcom innovador i útil.

Aquest treball final de carrera sorgeix doncs d'una idea molt ambiciosa: crear un sistema de processat de senyal digital que permetés, a temps real, convertir un timbre d'un instrument determinat a un altre, és a dir, a través d'un micròfon que capti el so d'un instrument, com ara el clarinet, analitzar i processar mitjançant programari d'ordinador aquest so d'entrada. Conseqüentment, convertir-lo per poder percebre'l a través dels altaveus com a so d'un altre instrument. D'aquesta manera, un músic amb habilitats per tocar el clarinet, mitjançant aquesta eina podria simular so de saxo, trompeta, etc., sense haver de recórrer a un sintetitzador ni a altres instruments MIDI.

Un concepte ben senzill, però un mètode que a dia d'avui encara no he vist. Potser serà per la complexitat que comporta? Bé, doncs la conclusió d'aquesta pregunta espero que es vagi desvetllant al llarg de l'estudi d'aquest treball.

Índex de Continguts

1	Introducció.....	5
1.1	Objectius.....	5
1.2	Introducció.....	5
1.3	Estructura de la Memòria.....	5
2	El So	7
2.1	Intensitat.....	7
2.2	To (<i>Pitch</i>).....	7
2.3	Timbre.....	9
2.4	Envolvents.....	11
2.5	Fase	15
3	El So Digital	16
4	Software	18
4.1	Introducció.....	18
4.2	Software de Càlcul Numèric	18
4.2.1	MatLab.....	18
4.2.2	Altres.....	19
4.3	Llenguatges de Programació “ <i>Live Coding</i> ”	19
4.3.1	CSound.....	19
4.3.2	SuperCollider.....	20
4.3.3	Chuck	21
4.4	Software amb Llenguatge de Programació Visual	21
4.4.1	PureData.....	21
4.4.2	MAX	22
4.4.3	Reaktor.....	23
4.4.4	SynC Modular.....	24
4.4.5	SynthMaker	24
4.5	Editors de So.....	25
4.5.1	SPEAR.....	25
4.5.2	Sonic Visualiser	25
4.5.3	Altres Editors de So.....	25
5	Filtres.....	27

6	Síntesi	34
6.1	Introducció	34
6.2	Mètodes de Síntesi	34
6.2.1	Síntesi Additiva	35
6.2.2	Síntesi Subtractiva	39
6.2.3	Síntesi per Modulació de Freqüències	40
6.2.4	Síntesi Granular	45
6.2.5	Síntesis Basades en Samples	46
6.2.6	Taula d'Ones	47
6.2.7	Síntesi per Modelatge Físic	48
7	Disseny	53
7.1	Introducció, Acotant el Sistema	53
7.2	Primer Disseny: Anàlisi del Clarinet i Suma Freqüencial	53
7.3	Segon Disseny: Utilitzant <i>Cross-Synthesis</i>	55
7.4	Tercer Disseny: Utilitzant MIDI	56
7.5	Quart Disseny: Utilitzant el Model d'Anàlisi/Síntesi Sinusoïdal + Soroll	57
7.5.1	Introducció a la Detecció de Pitch	57
7.5.2	Model d'Anàlisi / Síntesi Sinusoïdal + Soroll	58
7.6	Cinquè Disseny: Utilitzant Síntesi per Modelatge Físic	59
8	Conclusions	61
8.1	Conclusions	61
8.2	Treball Futur	62
9	Bibliografia	63

Índex de Figures

Fig. 1 Esquema de la relació entre to i altura.	7
Fig. 2 Espectre freqüencial d'una senyal d'àudio.....	10
Fig. 3 Diferents sons de clarinet. Representat per decibels al llarg del temps.	11
Fig. 4 Envoltent d'amplitud.	12
Fig. 5 Envoltent per a diferents dinàmiques.....	12
Fig. 6 Envoltent DSR.	13
Fig. 7 Envoltent sintetitzador casio CZ.....	14
Fig. 8 Exemple d'amplitud d'ona d'una nota interpretada amb guitarra.	14
Fig. 9 Exemple d'amplitud d'ona d'una nota interpretada amb flauta travessera.....	14
Fig. 10 Exemple de mostreig en senyals digitals.....	16
Fig. 11 Captura de l'entorn MatLab.	19
Fig. 12 Linies de codi <i>CSound</i>	20
Fig. 13 Captura de l'entorn SuperCollider.	20
Fig. 14 Exemple de sintaxis de <i>Chuck</i>	21
Fig. 15 Exemple de programació en <i>PD</i>	22
Fig. 16 Exemple entorn <i>MAX</i>	22
Fig. 17 Exemple entorn <i>Reaktor</i>	23
Fig. 18 Programa <i>SyncModular</i>	24
Fig. 19 Exemple entorn <i>SynthMaker</i>	24
Fig. 20 Interfície d' <i>SPEAR</i>	25
Fig. 21 Interfície de <i>Sonic Visualiser</i>	26
Fig. 22 Tipus de filtres.	27
Fig. 23 Filtre Passa-baix.	28
Fig. 24 Filtre Passa banda.	28
Fig. 25 Filtre Passa-alt.....	29
Fig. 26 De dalt a baix, parcial yA, yB i yC amb el guany aplicat.....	29
Fig. 27 Gràfica comparació espectre senyal original amb la equalitzada.	30
Fig. 28 Classificació tècniques de síntesis.	35
Fig. 29 Diagrama esquemàtic de la síntesis additiva. Les ones sinusoidals representades per freqüència <i>fk</i> i amplitud <i>rk</i>	35
Fig. 30 Diagrama detallat de la síntesi additiva.	37
Fig. 31 Forma d'ona resultant.....	38
Fig. 32 Espectre unilateral de <i>y(t)</i>	38
Fig. 33 Diagrama detallat de la síntesi substractiva.	39
Fig. 34 Ona portadora, o “ <i>carrier</i> ”.....	40
Fig. 35 Ona moduladora.	40
Fig. 36 Ona modulada.	41
Fig. 37 Ona complerta amb forma d'amplitud determinada per l'envoltent.	43
Fig. 38 Ampliació de forma d'ona resultant.	44
Fig. 39 Espectre unilateral de <i>y(t)</i>	44
Fig. 40 Exemples de diferents configuracions entre operadors de síntesi FM.	45

Fig. 41 Grànul sonor sinusoïdal.	45
Fig. 42 Sampler.	46
Fig. 43 Representació visual d'una taula d'ones amb l'emmagatzematge de les diferents formes.	47
Fig. 44 Esquema de la generació en bucle o looping.	48
Fig. 45 Esquema waveguide per a instrument similars al clarinet.	49
Fig. 46 Senyal de sortida sonora de 3 segons de duració on es veu la relació entre l'amplitud en db. del flux d'entrada en el que es basa el procés (verd), i la resultant (blau).	51
Fig. 47 Espectre unilateral de $y(t)$ amb freqüència fonamental 433 Hz.	51
Fig. 48 Espectre unilateral de $y(t)$ amb freqüència fonamental 123 Hz.	52
Fig. 49 Tessitura clarinet.	54
Fig. 50 Sistema d'Anàlisi/Síntesi per al model Sinusoïdal basat en McAulay & Quatieri.	58
Fig. 51 Esquema disseny utilitzant model de descomposició harmònica + soroll.	59
Fig. 52 Diagrama de blocs per a un disseny de transformació a partir de síntesi per modelatge físic.	60

1 Introducció

1.1 Objectius

L'objectiu principal d'aquest treball és fer un estudi de l'anàlisi del timbre i dels seus mètodes de síntesis per tal de dissenyar un sistema de transformacions tímbriques en instruments determinats.

1.2 Introducció

L'entorn de la tecnologia musical està en constant desenvolupament, i amb una bona base d'investigació científica al darrera, tant per empreses privades de so que ofereixen solucions per a la indústria, com universitats prestigioses, que dediquen cursos i molta recerca en aquests àmbits. A part del “boom” de la música electrònica, aquestes tècniques i procediments relacionats amb el maneig del so, sempre tindran efecte també a la música acústica o en definitiva tots els altres estils diferents de la música electrònica. Ja sigui per ampliar el ventall de recursos a nivell de producció, per perfeccionar gravacions, o tractar el so en directe, per exemple afegint efectes, distorsions, eliminar sorolls, o corregir desafinacions mitjançant sistemes d'auto-afinat vocal, etc. On també hi ha moltíssima investigació a darrere tant d'anàlisi espectral com de síntesis sonor és en l'àmbit del reconeixement i síntesis de veu, que cada cop és més essencial en aparells electrònics, ja sigui per interactuar amb ordres, o redactar directament a partir de text dictat, o per simular veus virtuals, o facilitar problemes en discapacitats...

La finalitat de modificar senyals de so, en aquest cas d'un instrument determinat, per obtenir-ne d'altres amb unes característiques concretes m'obliga a fer un estudi sobre el so d'aquests instruments que es veuen implicats, sobre els mètodes de síntesis utilitzats per tal de simular-los de manera digital, conèixer l'àudio digital, el medi en el qual es podrien dissenyar filtres necessaris per aquest procés, el programari necessari per prendre mostres, analitzar-les i dur a terme aquests filtratges i modificacions, entre d'altres conceptes previs ja siguin relacionats amb el món de l'acústica i la física del so, de la informàtica musical, o del tractament i processat de la senyal.

1.3 Estructura de la Memòria

L'estructuració d'aquesta memòria es basa en l'ordre en que s'han estudiat els conceptes teòrics per tal de poder abordar els objectius marcats.

En el capítol 1 s'introdueix el projecte detallant també quines motivacions i objectius han sigut els que l'han impulsat.

En el capítol 2 es defineixen alguns conceptes teòrics bàsics per al tractament de senyals d'àudio. Per tal de dur a terme una modificació tímbrica es fa una gran incisió amb el so, i el seu timbre

En el capítol 3 s'introdueix l'àudio digital. Per dur a terme un estudi i maneig amb el so a través de l'ordinador, es necessita introduir prèviament l'àudio digital, i veure el paper que tenen junt amb l'àudio analògic.

En el capítol 4 s'exposa un ampli ventall de programari útil per analitzar, crear i en definitiva treballar des del medi digital amb tot tipus d'aspectes en l'àmbit d'aquest treball.

En el capítol 5 s'introdueixen un dels elements clau en el procés de la modificació de la senyal, els filtres, en aquest cas els digitals. S'exposa els tipus de filtres que hi ha i el seu funcionament.

En el capítol 6 es detallen els principals mètodes de síntesi ja que juntament amb el concepte d'anàlisi i de modificació mantenen una relació molt estreta. Primer s'estudien amb detall els principals mètodes i se n'extreuen exemples útils tan per als processos de creació de nous timbres, com per a la modificació d'altres ja existents, descobrint doncs com d'interessant poden ser aquestes tècniques utilitzades en el procés de modificació tímbrica.

En el capítol 7 es dissenyen varies possibilitats d'algorismes de modificació de timbres determinats. Tot i els conceptes prèviament vistos es necessitarà entrar en altres detalls com:

- Introducció dels possibles dissenys i acotacions del sistema.
- Un petit anàlisi del clarinet i el seu timbre, que ha sigut l'instrument elegit com a principal font sonora i així doncs el timbre al qual aplicar-hi transformacions.
- La introducció del MDI. Què és, el lligam que te amb la síntesi de so, i com actua dins del possible disseny.
- Els mètodes de detecció de pitch, o estimació de freqüència instantània.

En el capítol 8 s'exposen les conclusions finals, la feina que ha quedat pendent i les possibles ampliacions o intencions futures.

En el capítol 9 s'inclou la bibliografia, d'on s'ha fet referència, extraient conceptes, idees i exemples.

Tot seguit, al final del document s'inclou el grup d'annexos.

2 El So

El so es la variació periòdica de pressió que genera un objecte i que es propaga per un medi. Quan les vibracions es produeixen de manera arbitrària diem que es tracta d'un soroll.

Entre les característiques fonamentals per definir un so trobem la durada, l'altura, la intensitat, i el timbre. A continuació es detallen les més influents en aquesta recerca:

2.1 Intensitat

Depèn de l'energia que transporta l'ona sonora i està en relació directa amb la seva amplitud. Una ona de més amplitud provocarà en nosaltres una sensació de so més fort. Es mesura en decibels (dB).

2.2 To (*Pitch*)

El to és la qualitat del so mitjançant la qual l'oïda li assigna un lloc en l'escala musical, permetent distingir entre els greus i els aguts. La magnitud física que està associada al to és la freqüència i, encara que entre els dos termes existeixi una relació molt estreta, no es refereixen al mateix fenomen.

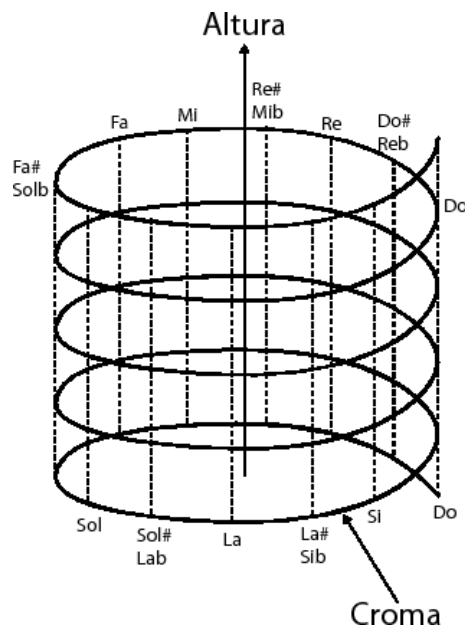


Fig. 1 Esquema de la relació entre to i altura.

El to és una magnitud subjectiva i es refereix a l'altura o gravetat d'un so. En canvi, la freqüència és una magnitud objectiva i mesurable referida a formes d'ona periòdiques.

El to d'un so augmenta amb la freqüència, però no en la mateixa mesura. Gràcies a ella es mesura el nombre de vibracions. La seva unitat de mesura és l'Hertz (Hz) que es refereix a tantes vibracions per segon. Per tant, una freqüència d'1 Hertz és el mateix que dir que el so té una vibració per segon.

Els sons percebuts com a greus corresponen a freqüències baixes, mentre que els aguts són deguts a freqüències altes. Per tenir un exemple, el so més greu d'una guitarra correspon a una freqüència de 82,4 Hz i el més agut a 698,5 Hz.

Per calcular la relació exacta que mantenen les notes amb la freqüència, partim de que el nostre sistema musical té 7 notes o tons, però si les separem en semitons en tenim 12, que són els espais horitzontals de la figura anterior anomenats cromes. Aquestes 12 notes es repeteixen octava rere octava i la relació entre octaves és el doble de freqüència per a la superior i la meitat per a la inferior. Així doncs un La3 a 440 Hz que és l'estàndard recomanat internacionalment per afinar els instruments, es converteix en un de 880 Hz quan es toca una octava més amunt, llavors anomenat La4.

Aquesta relació deixa clar que no es tracta d'una relació lineal, ja que entre el La2 i el La3 tenim una diferència de 220 Hz, i en canvi entre el La3 i el La4 una de 440 Hz, però en canvi s'observa com hi ha una proporció constant i logarítmica, és a dir que varia exponencialment.

Per introduir la raó r , o el factor multiplicatiu constant a partir del La3 que sabem que és de 440 Hz, i obtenir el següent cromatisme, el La sostingut notat amb el símbol $\#$, es diu que:

$$\mathbf{La\#3 = 440 \cdot r}$$

I si es continua la sèrie per buscar el següent, el Si3:

$$\mathbf{Si3 = 440 \cdot r \cdot r = 440r^2}$$

Amb això, es pot arribar a la conclusió de que la octava, La4, serà:

$$\mathbf{La4 = 440 \cdot r^{12}}$$

I a partir d'aquí, com que ja es coneix que en octaves es doblen freqüències, i el La4 equival a 880 Hz, s'obté el valor exacte d'aquesta raó:

$$\mathbf{440 \cdot r^{12} = 880}$$

$$\mathbf{r^{12} = 2}$$

$$\mathbf{r = \sqrt[12]{2}}$$

$$\mathbf{r = 1,059463}$$

Així doncs, ja es pot obtenir qualsevol freqüència a partir d'una nota coneguda sabent la distància d , que la separen en semitons:

$$F_2 = F_1 \cdot r^d$$

Cal apuntar que també serveix per trobar freqüències en semitons inferiors, es a dir, distàncies negatives. Per exemple a partir del *La3* a 440 Hz si es vol obtenir el *Sol2*, situat 14 semitons per sota:

$$\text{Sol2} = 440 \cdot 1,059463^{-14} = 195,998 \text{ Hz}$$

Aquesta raó doncs, permet complimentar la taula afegida a l'**Annex A** amb el llistat de notes, amb les seves freqüències i longitud d'ona corresponent, les quals s'utilitzaran com a referent al llarg de la realització d'aquest treball.

Tot i poder representar infinitud de notes amb les seves freqüències, no totes les ones sonores poden ser percebudes per l'oïda humana, la qual és sensible únicament a aquelles freqüències compreses entre els 20 i els 20.000 Hz. A l'aire aquests valors extrems corresponen a longituds d'ona que van des de 16 metres fins a 1,6 centímetres, respectivament i en general es tracta d'ones de petita amplitud.

Juntament amb la freqüència, intervenen altres factors de caràcter psicològic en la percepció sonora del to. Així succeeix que, en elevar la intensitat s'eleva el to percebut per freqüències altes i es baixa per a les freqüències baixes. El to és relativament independent de la intensitat en freqüències compreses entre 1000 i 3000 Hz.

2.3 Timbre

El timbre és la qualitat del so que permet distingir sons procedents de diferents instruments, fins i tot quan tenen el mateix to i intensitat. Degut en aquesta mateixa qualitat és possible reconèixer a una persona per la seva veu, que resulta característica de cada individu.

Aquest fenomen és degut a que un so no està format només d'una freqüència, sinó per la suma d'altres que son múltiples de la fonamental. Aquestes altres freqüències varien en intensitat i son anomenades harmònics. La proporció i intensitat d'aquests harmònics son diferents en cada instruments i per això en podem diferenciar els seus sons.

Jean Joseph Fourier va demostrar matemàticament que tota ona periòdica amb freqüència f pot ser descomposta en una quantitat infinita d'ones sinusoïdals de freqüències $f, 2f, 3f, 4f \dots$

Les sinusoïdals manquen d'harmònics, per la qual cosa no es poden descompondre més i és consideren pures. Aquest mètode de descompondre una senyal és conegut com anàlisi de Fourier.

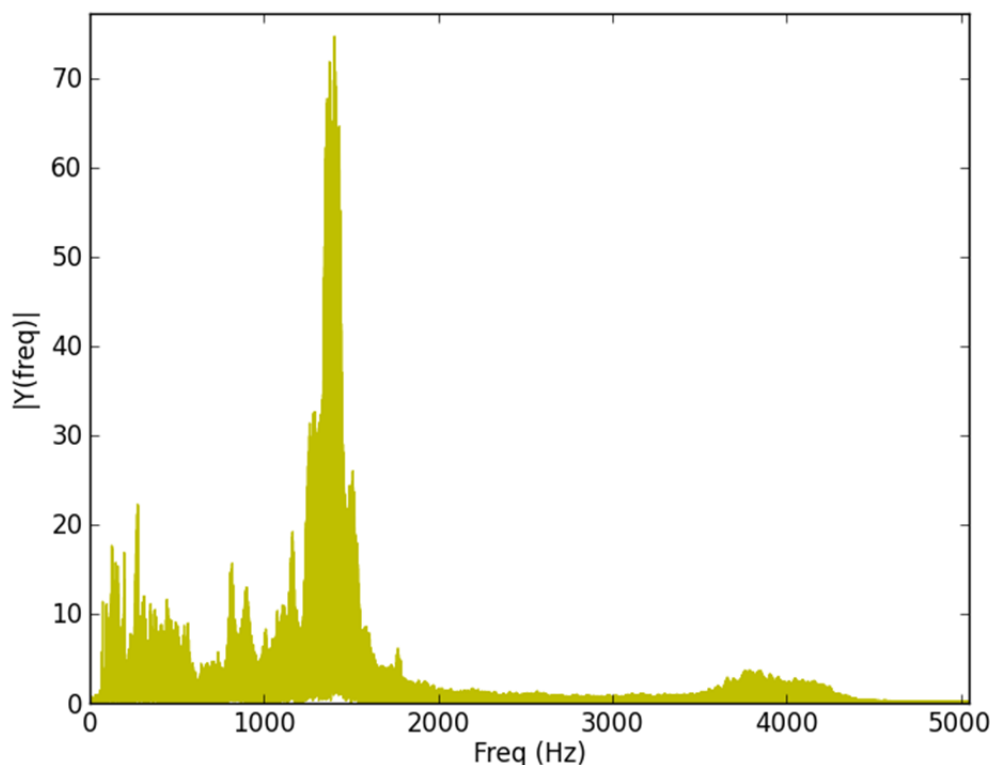


Fig. 2 Espectre freqüencial d'una senyal d'àudio.

Si a una senyal se li van afegint harmònics, la forma d'ona anirà canviant però la seva freqüència fonamental romandrà inalterada. Per tant veiem que el timbre varia en raó dels harmònics mentre que el to es manté. Cal destacar que la posició en freqüència dels harmònics no és aleatòria. Els harmònics es situen en múltiples enters de la freqüència fonamental. Es a dir:

$$f_a = i * f_f$$

essent f_a freqüència de l'harmònic, i el numero de harmònic i f_f la freqüència fonamental.

I consegüentment la longitud d'ona manté la proporció inversa.

Les amplituds relatives de cada harmònic varien en funció de la forma d'ona, essent normalment el de major amplitud el que es considera fonamental. Per aquest motiu el timbre està relacionat amb la complexitat de les ones sonores que arriben a l'oïda. Els diapasons son capaços de generar sons purs, degut a la seva única freqüència emesa, però els instruments musicals, contràriament, donen lloc a un so més ric que resulta de les vibracions complexes, augmentant el numero d'harmònics. Cada vibració complexa pot considerar-se composta per una sèrie de vibracions harmòniques simples d'una freqüència i d'una amplitud determinades, cada una de les quals, si es considera separatament, esdevindria un so pur. Aquesta mescla de tons parcials és una característica de cada instrument i defineix el timbre.



Fig. 3 Diferents sons de clarinet. Representat per decibels al llarg del temps.

En la figura anterior s'observa dues gravacions d'una mateixa nota de clarinet. Gràcies a les marques temporals es pot apreciar que les dues gravacions tenen la mateixa freqüència (on en 10ms tenim uns 5 cicles, que equivaldria a 500Hz), però la seva forma d'ona o timbre percebut és diferent tot i estar interpretat pel mateix instrument. Aquest és un clar exemple de com el timbre depèn de molts factors, i en aquest cas, aquesta mateixa nota es va gravar en diferents espais, afectant doncs l'acústica de la sala a la reflexió dels harmònics enregistrats pel micròfon. Altres aspectes que poden fer variar l'estructura harmònica a l'hora de tocar un instrument de vent són alguns paràmetres d'interpretació com ara la pressió que es fa amb la boca sobre l'embocadura, la intensitat en la que toca l'interpret o la temperatura ambient que afecta notablement a la física de l'instrument per exemple amb petites dilatacions.

Així doncs com que de timbre en un mateix instrument no n'hi ha cap d'identíc, a l'hora de modelar-lo s'intenta buscar un so bastant correcte amb els paràmetres que es creuen oportuns per a que soni el més estàndard possible.

2.4 Envolvents

Una altra de les característiques del so o senyal musical, a part del to i el timbre, és la seva duració, i la variació de la intensitat al llarg d'aquesta, anomenat envoltent. L'envoltent ADSR representa aquest esquema estàndard que determina les principals fases de variació d'amplitud, o intensitat de la senyal durant el seu transcurs.

El so en general, i en aquest cas els harmònics que conjuntament formen una nota d'un instrument, tenen comportaments independents (tot i que es puguin assemblar entre ells) en el temps. Això significa que cada parcial harmònic recorre una trajectòria envoltent particular en la seva evolució temporal. I el so com a un tot, també té un comportament definit per un envoltent d'amplitud tal i com es mostra a la *figura 4*.

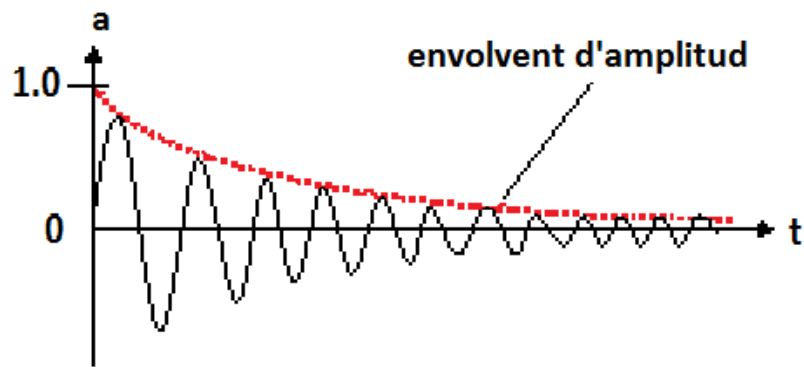


Fig. 4 Envolvent d'amplitud.

La *figura 5* mostra la forma de l'envolvent de sons reals per diferents dinàmiques.

L'evolució temporal del so és clarament diferent depenent del nivell d'energia, definida pel seu contingut de freqüències presents en la senyal acústica.

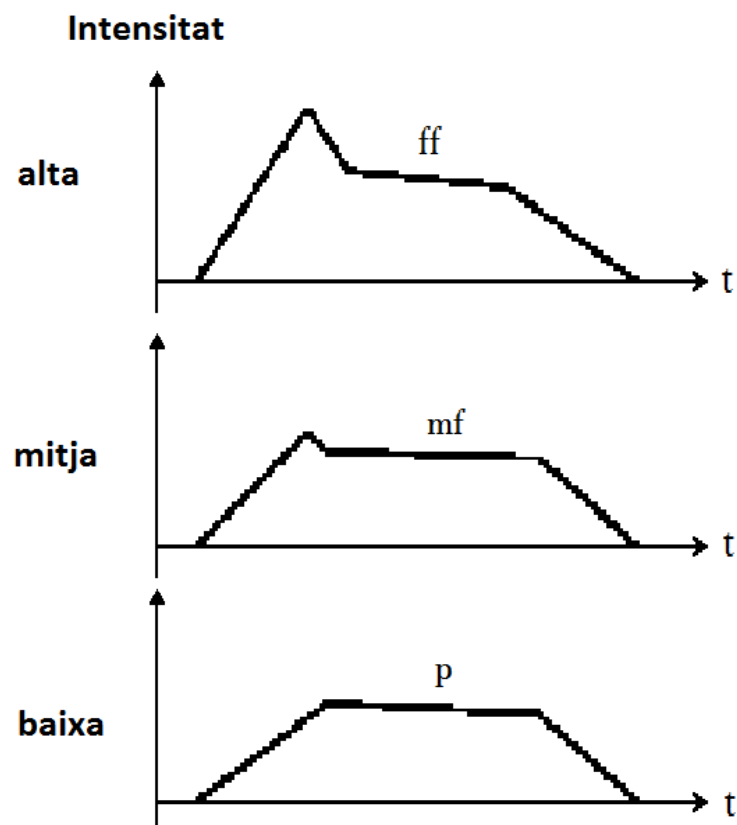


Fig. 5 Envolvent per a diferents dinàmiques.

Existeixen moltes formes d'envolvent, però generalment segueixen un patró o evolució comú. La forma més genèrica d'envolvent es denomina ADSR (atac, decaiguda, sosteniment, relaxació) per l'abreviatura de les seves diferents parts.

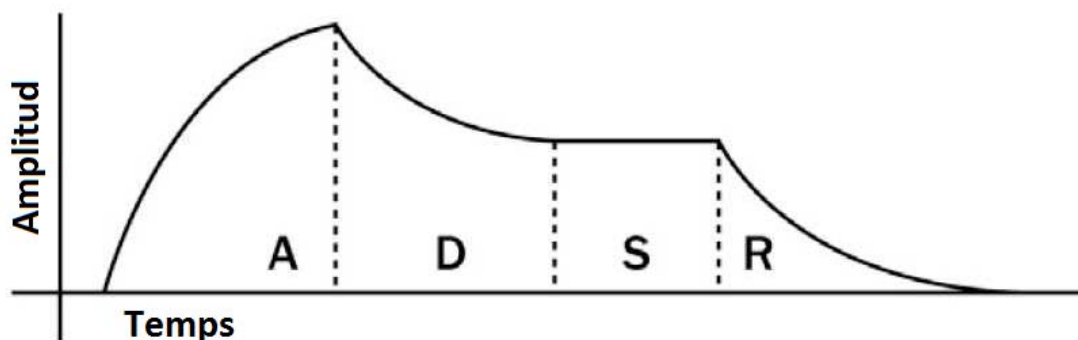


Fig. 6 Envolvent DSR.

- Atac (A - Attack). És el temps que tarda la senyal sonora des de l'inici en arribar al seu màxim d'amplitud.
- Decaiguda (D - Decay). És el temps que tarda en establir-se la senyal sonora un cop ja ha fet el pic màxim d'amplitud.
- Sosteniment (S - Sustain). El temps que dura la senyal sonora de forma estable. Aquest estat del so, és el que el músic pot controlar depenent de l'instrument, i en funció de com manté la durada i intensitat de la nota.
- Relaxació (R - Release). El temps que dura la senyal sonora en atenuar-se i finalitzar.

En el cas dels sintetitzadors els envoltants usualment s'utilitzen per controlar l'amplitud de sortida d'un oscil·lador. Una vegada triada la forma de l'envolvent, aquesta s'escala tant en temps com en amplitud i es multiplica per la senyal de l'oscil·lador. D'aquesta manera es possible manipular el comportament temporal de la senyal sintetitzada.

En la *figura 7* es pot observar una envoltant més complexa, en aquest cas del sintetitzador casio CZ, la qual permet parametritzar més fases per donar un so més realista o personalitzat.

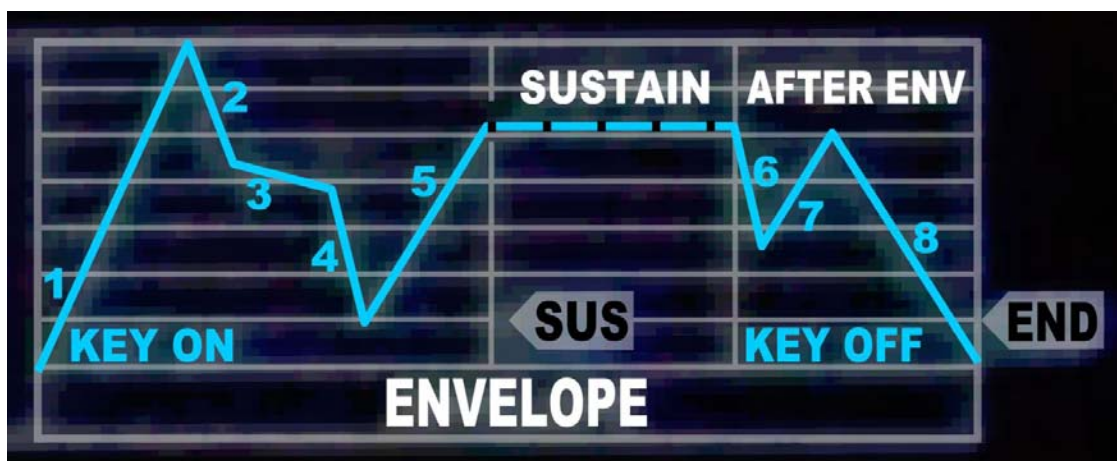


Fig. 7 Envolvent sintetitzador casio CZ.

Està demostrat que en els instruments reals es produeix una gran variació tímbrica en aquests diferents estats de la vida de la nota interpretada. Només a la part del sosteniment és on es pot considerar un major sosteniment també en quant a components harmònics. Això significa que quan s'ataca una nota, des de que l'amplitud de potencia sonora és nul·la fins que arriba al seu pic màxim, que equivaldria al parcial d'atac de l'envolvent, el timbre canvia considerablement. I el mateix passa en la decaiguda o relaxació, on no només hi ha pèrdua de potencia, sinó que en la majoria d'instruments hi ha un gran canvi de configuracions harmòniques, o sigui, variacions tímbriques.

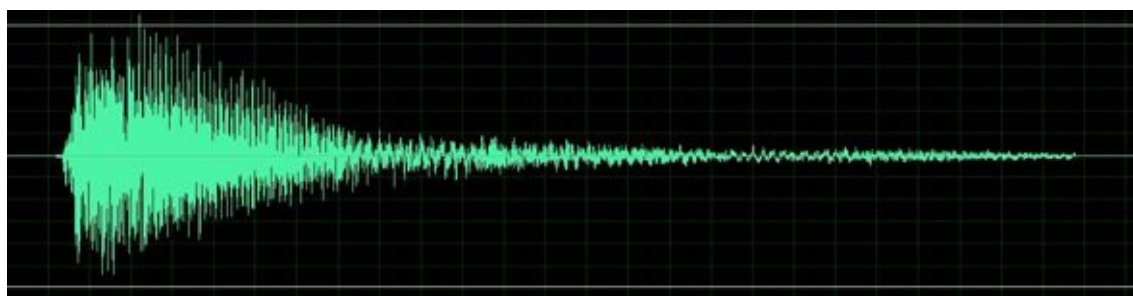


Fig. 8 Exemple d'amplitud d'ona d'una nota interpretada amb guitarra.

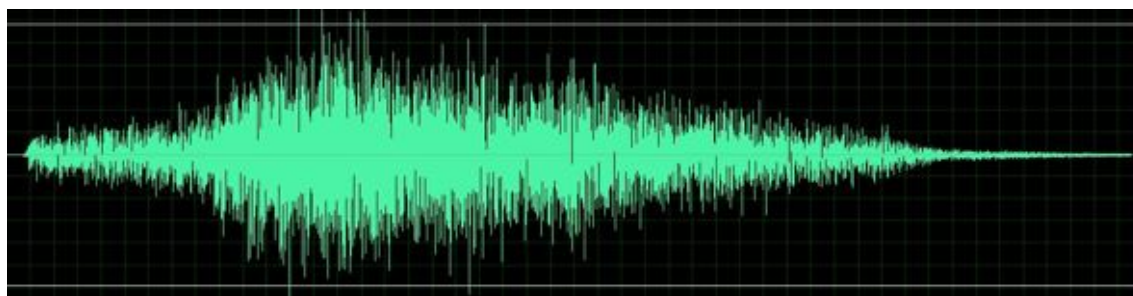


Fig. 9 Exemple d'amplitud d'ona d'una nota interpretada amb flauta travessera.

Es per aquest motiu que en aquest treball es plantejarà seleccionar instruments musicals amb envoltants similar, com podrien ser dos instruments de la família de vent, i d'aquesta manera centrar l'estudi en la part central o *sustain* que és on més constants són les característiques tímbriques.

2.5 Fase

Es defineix com a fase al desplaçament o desfasament d'una senyal respecte a un punt de referència determinat de forma arbitrària. La fase es mesura en radians, amb el rang $[0, 2\pi]$ o en graus, amb el rang $[0, 360]$

Una fase més gran a 2π o 360 manca de sentit físic, ja que no és possible distingir de una que esdevé dins del rang normal. La fase d'un so aïllat no altera la seva percepció, agafa importància quan dos o més sons es mesclen entre sí. Per exemple, dos sons poden ser idèntics, però estar desfasats entre sí, cosa que implica que un so comenci abans que el altre en mesures inapreciables com a oients, però al interactuar, el resultat percebut pot canviar radicalment dependent del grau de desfasament entre ells. Si el desfasament és 0 o 2π , els sons al mesclar-se es sumen, i com que les zones de rarefacció i compressió d'ambdós sons coincideixen, com a resultat s'obté el mateix so però amplificat. En canvi si el desfasament és de 2π o 180 graus, significa que les zones de rarefacció d'un so coincideixen amb les de compressió de l'altre i al mesclar-se, els sons s'anul·len completament. El resultat és que no es percep res, però això no és un problema de percepció, sinó que és un fenomen purament físic.

La fase és una variable bastant ignorada i en general es té poc en compte, però és sumament important en la anàlisi de Fourier i algunes tècniques de síntesis. Podem veure un clar exemple a l'hora de comparar senyals i és que dues funcions iguals amb mateixos harmònics i amplituds, però amb una petita variació de fase en algun dels seus harmònics secundaris, poden generar unes formes d'ona completament diferents, i en canvi ser pràcticament indistingibles entre elles a nivell auditiu. Per aquest motiu és tant important la anàlisi espectral a l'hora de comparar les senyals.

3 El So Digital

Per tal de poder emmagatzemar l'àudio analògic en format digital necessitem fer una conversió, com també necessitem un altre tipus de conversió quan volem reproduir un so digital. Aquests son alguns de les etapes del processament del senyal com també ho serien les transformacions d'aquestes, l'emmagatzematge o la transmissió.

Els sistemes digitals son discrets en el temps i nivells, i a l'hora de convertir una analògica a digital, es mostreja per freqüència. Cada diferent valor de mostreig s'emmagatzema en un numero de bits determinat, es a dir, es discretitza.

La freqüència de mostreig, és el número de vegades per segon que es pren una mostra, i el numero de bits utilitzat, o altrament anomenat profunditat de bit, és el que determina el numero de valors digitals diferents que pot tenir cada mostra. Evidentment com major és el mostrejat i el número de bits utilitzat, més acurada serà la forma d'ona a representar.

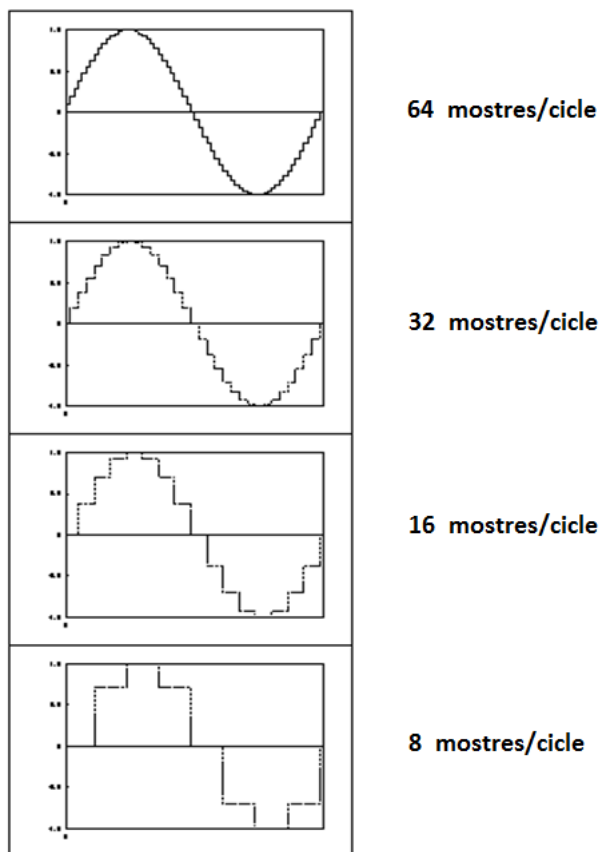


Fig. 10 Exemple de mostreig en senyals digitals.

El tractament digital és molt utilitzat per enregistrar, editar i distribuir el so entre d'altres per la facilitat en detecció i correcció d'errors i perquè el sistema analògic és molt més susceptible en quant a soroll i distorsió. Per altra banda un aspecte negatiu de digitalitzar el so és la limitació en el rang de freqüència màxima.

Com el teorema de Nyquist indica, només podem representar freqüències que tinguin un valor igual o inferior a la meitat de la freqüència de mostreig. I és que sinó, es pot donar el cas de que es malmeti la informació o hi hagi pèrdues de dades, per exemple, quant les mostres es prenguin en el punt d'amplitud 0, es a dir quan l'ona de so es creua amb l'eix i així doncs quedant el so totalment anul·lat.

Existeix un ús molt ampli de diferents filtres que es poden utilitzar en els sistemes d'àudio digitals. Ja sigui per implementar funcions a fi de controlar tons greus i aguts de l'estèreo, o bé per eliminar sorolls no desitjats que disminueixen la qualitat del senyal que es percep. Aquests sorolls estan causats per senyals d'alta freqüència que poden tenir diferent origen:

- Es poden injectar al dispositiu d'àudio des de fons externes, com és el cas de les imperfeccions mecàniques dels micròfons.
- Degut als transitoris de canvi de fase dels semiconductors dels components electrònics. L'ànim de qualsevol equip de so és proporcionar una bona qualitat sonora, per això es fa ús de filtres com a part del tractament digital per tal d'eliminar aquests components no desitjats.

Un dels procediments pràctics d'aquest treball es basarà en el disseny d'un sistema de filtrat d'àudio. Aquest haurà de ser capaç d'eliminar els components de freqüències no desitjades del senyal entrant, i d'aquesta manera aconseguir un so diferent, filtrat de manera digital.

4 Software

4.1 Introducció

Per tal de poder adquirir i analitzar dades, dissenyar filtres, mostrar gràfiques o també d'intentar implementar una aplicació final amb les funcionalitats desitjades, s'exposen algunes eines de càlcul, i llenguatges de programació orientats a la manipulació del so, adients per duar a terme les tasques i objectius plantejats.

Aquests softwares s'han classificat en diferents grups. El primer es basa en programes numèrics que permeten realitzar càlculs i contenen nombroses funcions matemàtiques. El segon grup és de llenguatges de programació "*Live Coding*", es a dir, codi que es pot modificar en temps d'execució, i específics per treballar amb so. El tercer grup també orientat al so, és d'entorns amb llenguatge de programació visual o VPL, es a dir, que permet programar mitjançant la manipulació d'elements gràfics; software utilitzat en l'**Annex D** per a una comparativa de síntesi sonora detallada al capítol 6 d'aquesta memòria. I l'últim grup consta d'una petita llista d'aplicacions amb funcionalitats concretes útils per analitzar i editar so.

4.2 Software de Càlcul Numèric

4.2.1 *MatLab*

Abreviatura de MATrixLABoratory, és un entorn IDE d'altres prestacions per la realització de càlculs tècnics o científics. El paquet integra mòduls de computació, visualització i programació en un entorn relativament fàcil d'utilitzar. Per posar alguns exemples direm que MatLab inclou funcions per càlcul matemàtic, desenvolupament d'algorismes, adquisició i presentació de dades, modelatge, simulació, creació de prototips, creació d'interfícies d'usuari, etc. En aquest projecte ens interessa particularment per les eines que disposa de processament de senyals digitals.

MatLab és un sistema interactiu en el qual l'element bàsic és una matriu que no requereix dimensionat previ. Tot i que en llenguatges no interactius com C o Fortran el temps d'execució del codi sigui inferior a *MatLab*, aquest últim permet escriure aquest codi en molt menys temps, especialment si apareix formulació, matrius i vectors. Tot això fa que MatLab es presenti com una eina ideal en la fase d'anàlisi, disseny, desenvolupament i proves per aquest projecte.

Opcionalment, *MatLab* també disposa de toolboxes y SIMULINK® amb els pertinents blocksets. Les toolboxes son biblioteques de programes que permeten estendre la capacitat de treball de *MatLab* en diverses àrees específiques, com control de sistemes, processat de senyal, càlcul simbòlic, estadística, comunicacions... I el software SIMULINK utilitza una interfície gràfica que permet modelar, simular i analitzar sistemes dinàmics a un nivell de diagrama de blocs. Aquests sistemes poden ser tan lineals com no, continus o discrets.

Cal concretar que per treballar amb processament de la senyal, com en aquest cas, és molt recomanat utilitzar el *signal processing toolbox*.

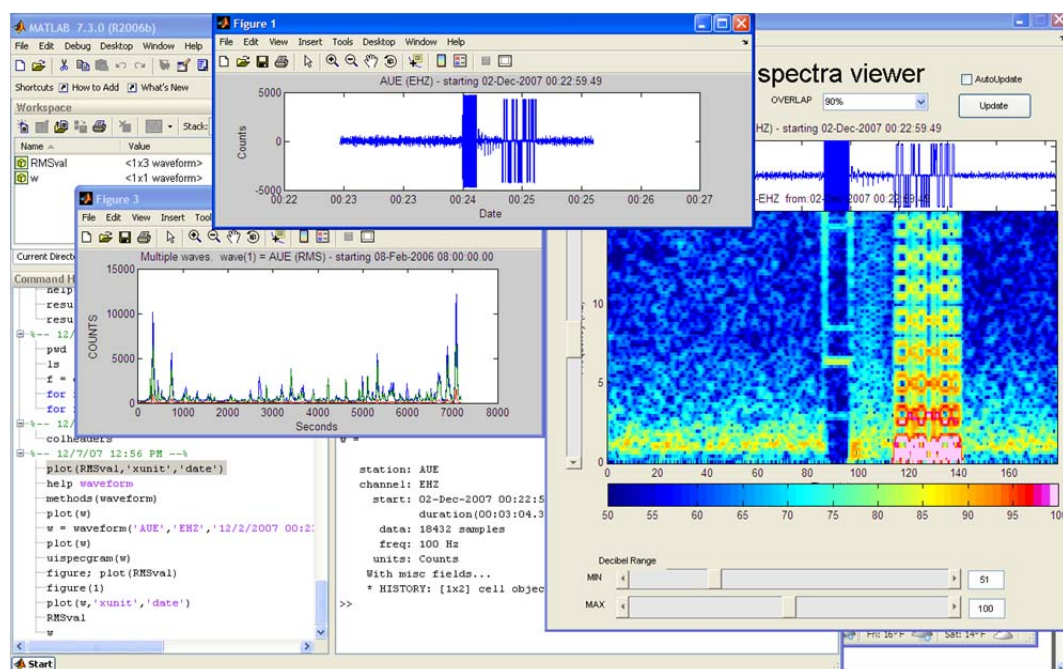


Fig. 11 Captura de l'entorn MatLab.

4.2.2 Altres

Entre altres alternatives a *MatLab* en software lliure hi trobem *Octave*, que forma part del projecte GNU, i el llenguatge del qual també és orientat a anàlisi numèrica amb una sintaxis pràcticament igual que la de *MatLab*. O *Scilab*, també lliure, amb més anys de funcionament que *Octave*, i també amb una sintaxis fonamentada en *MatLab*.

4.3 Llenguatges de Programació “Live Coding”

4.3.1 CSound

És un llenguatge de programació orientat a la composició musical i la generació de so tot permetent als usuaris programar els seus propis mètodes de síntesis i així doncs els timbres desitjats.

CSound és de codi obert, no requereix molts recursos hardware, i és possible utilitzar-lo en temps real, crear interfícies d'usuari, i interaccionar amb altres programaris externs.

El seu compilador està programat amb llenguatge C, d'aquí el seu nom, i el seu funcionament es basa en una estructura d'orquestra amb els diferents instruments i tot això copsat en un patró, comparable en una partitura, que s'acaba convertint en el fitxer predeterminat per a *CSound*.

```

<CsoundSynthesizer>
<CsOptions>
</CsOptions>
<CsInstruments>

sr = 44100
ksmps = 128
nchnls = 2
0dbfs = 1

instr 1
kFreq line 100, 5, 1000 ; 'line' generates a linear ramp, from 100-1000 Hz, taking 5 seconds
aOut oscili 0.2, kFreq, 1 ; an oscillator whose frequency is taken from the value produced by 'line'
outvalue "freqsweep", kFreq ; show the value from 'line' in a widget
outs aOut, aOut ; send the oscillator's audio to the audio output
endin

instr 2
kFreq expon 100, 5, 1000 ; the 'expon' exponential curve is more useful when working with frequencies
aOut oscili 0.2, kFreq, 1
outvalue "freqsweep", kFreq
outs aOut, aOut
endin

</CsInstruments>
<CsScore>
f 1 0 1024 10 1 ; the basic sine waveform for the oscillator is generated here
i 1 0 5
i 2 5 5 ; the exponential curve goes more even though the octaves
e
</CsScore>
</CsoundSynthesizer>

```

Fig. 12 Linies de codi *CSound*.

4.3.2 SuperCollider

Igual que *CSound*, *SuperCollider* és un entorn de programació amb llenguatge propi orientat a objectes i enfocat a la síntesis de so en temps real i a la composició algorísmica.

Es un projecte de codi obert i multi-plataforma, utilitzat per músics, científics i artistes que treballen amb els so del qual se'n destaca la seva flexibilitat i operativitat en temps real per sobre de *CSound*.

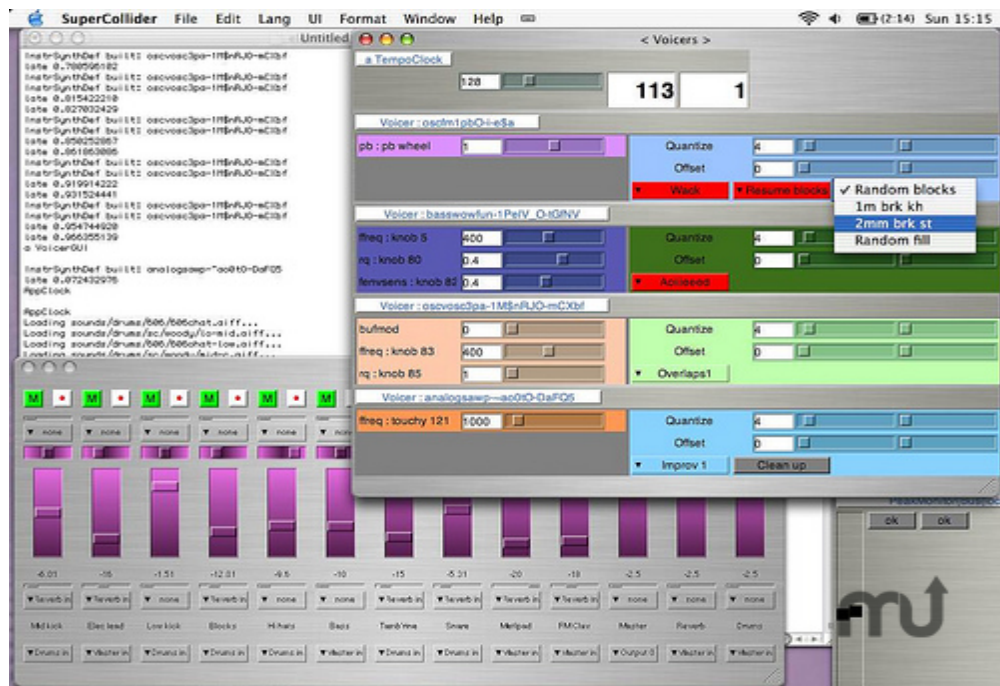


Fig. 13 Captura de l'entorn *SuperCollider*.

4.3.3 Chuck

Chuck és també un llenguatge multi-plataforma enfocat a la creació sonora, però més fàcil i modern que els dos anomenats anteriorment. En destaca un model de programació concurrent, precís i expressiu, que permet afegir i modificar codi en funcionament, *on-the-fly*, com si es tractés d'una interpretació en viu d'un instrument musical. Es per aquest motiu que ja es troben algunes orquestres o grups musicals que improvisen concerts amb sons experimentals creats per ordinador mitjançant *Chuck*.

```
107 //
108 // BD - bass drum
109 class BD_808 extends Chubgraph
110 {
111     inlet => BridgedT bt => LPF lp => HardLimit hl => outlet;
112
113     16000 => lp.freq; // tuned by ear
114
115     47.444 => float freq; //
116     4.0 => float decay;
117
118     2.0 => bt.gain; // tuned by ear
119     bt.setFreq(freq); bt.setDecay(decay);
120
121     fun void noteOn(float accent) // accented when noteOn is passed a float > 0.0
122     {
123         if (accent != 0.0)
124         { // if accent, do trick as described in [1]
125             bt.setFreq(freq*2.813; // up an octave
126             bt.noteOn();
127             4::ms => now;
128             bt.setFreq(freq); // fundamental
129         } else { noteOn(); } // call unaccented version
130     }
131
132     fun void noteOn() // unaccented when noteOn is passed nothing
133     {
134         bt.setFreq(freq);
135         bt.noteOn();
136     }
137 }
```

Fig. 14 Exemple de sintaxis de *Chuck*.

4.4 Software amb Llenguatge de Programació Visual

4.4.1 PureData

També abreviat com a *PD*, és un llenguatge programació lliure i multi-plataforma basat en entorn gràfic i en temps real, amb sintaxis relativament senzilla, dedicat al processament digital de senyals i sons electrònics mitjançant flux de dades. És popularment utilitzat en disseny d'algorismes musicals, tractament sonor, i processos de síntesis, ja que afavoreix la creativitat en l'exploració i experimentació de nous conceptes sonors.

Així doncs les característiques principals de *PD* son les de generar processos d'àudio, vídeo i gràfics en temps real, de manejar sensors físics, de interactuar amb el protocol MIDI o de comunicar-se amb altres aplicacions o plataformes.

A més a més, *PD* està dissenyat nativament per a desenvolupar projectes col·laboratius en viu a través de xarxes o d'internet, permetent a músics interconnectats crear música junts en temps real.

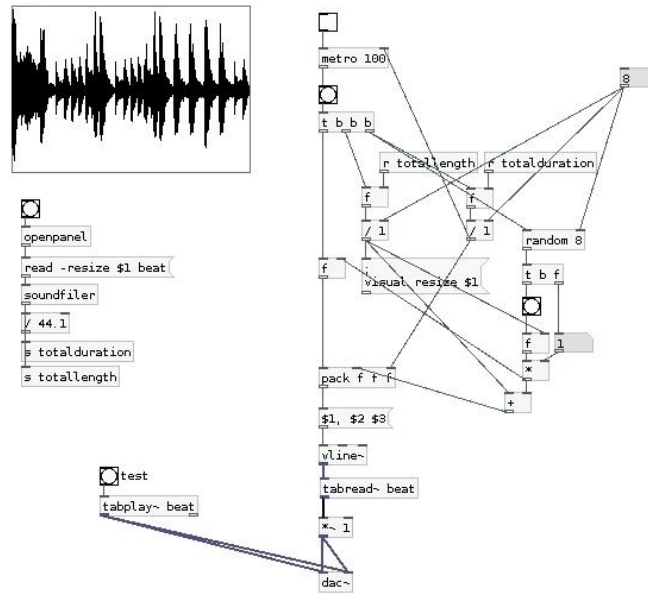


Fig. 15 Exemple de programació en *PD*.

4.4.2 *MAX*

MAX és un entorn de desenvolupament basat en un llenguatge de programació visual y de codi propietari suportat per l'empresa *Cycling '74*. El programa, amb més de vint anys de vida, està enfocat a compositors, artistes i dissenyadors d'altres programes, interessats en la creació d'interfícies musicals interactives.

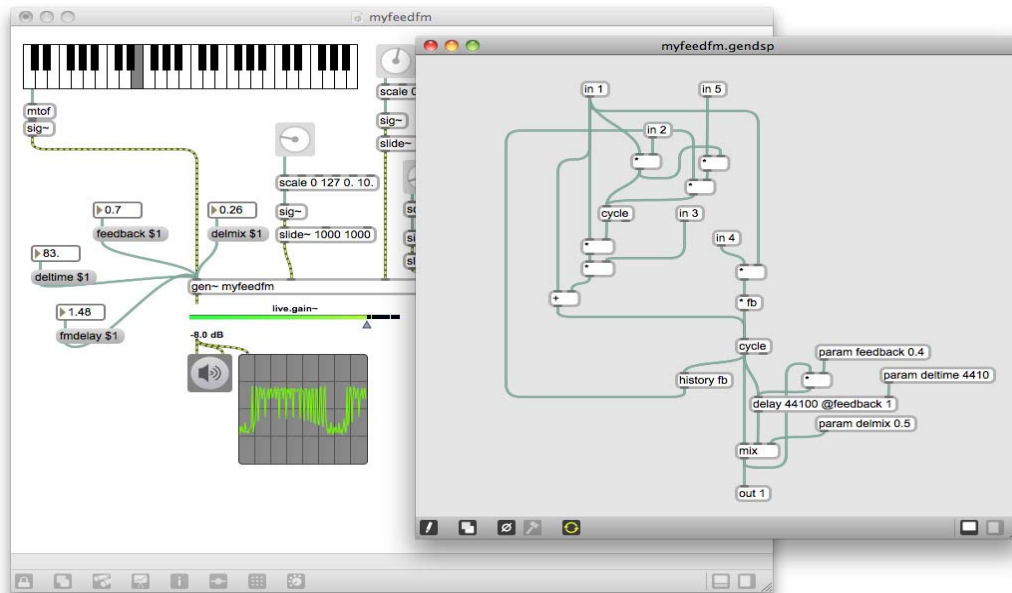


Fig. 16 Exemple entorn *MAX*.

Son molt similars amb *PureData*, però per molts professionals, *MAX* és considerat el més poderós d'entre els softwares que abasten aquest àmbit. És modular tot permetent interoperabilitat amb altres softwares externs com *PureData* gràcies a l'API pública que connecta els processos. També conté biblioteques de codi obert, i interfícies per a crear aplicacions externes que per exemple es poden afegir com a *plugin* d'altres entorns de composició musical.

4.4.3 Reaktor

És un programa propietari que implementa un estudi de música modular i gràfic. Desenvolupat per *Native Instruments*, permet als enginyers i músics dissenyar i construir els seus propis instruments, samplers, efectes i eines de disseny sonor. Amb ell venen inclosos molts sintetitzadors i instruments ja implementats i llestos per utilitzar, com ara emulacions de sintetitzadors analògics clàssics, o d'altres de futuristes que permeten un disseny sonor de forma interactiva.

A més dels 70 sintetitzadors exclusius, generadors de so, caixes de ritmes, seqüenciadors i efectes que venen amb el programa cal sumar-li els 3000 de disponibles de forma gratuïta a les llibreries online alimentades pels mateixos usuaris. Però la part més interessants, és que tots aquests instruments poden ser examinats i modificats lliurement, en funció de la versió adquirida, fomentant doncs la enginyeria inversa, i permetent ampliacions i connexions amb softwares externs gràcies a l'arquitectura plenament modular.

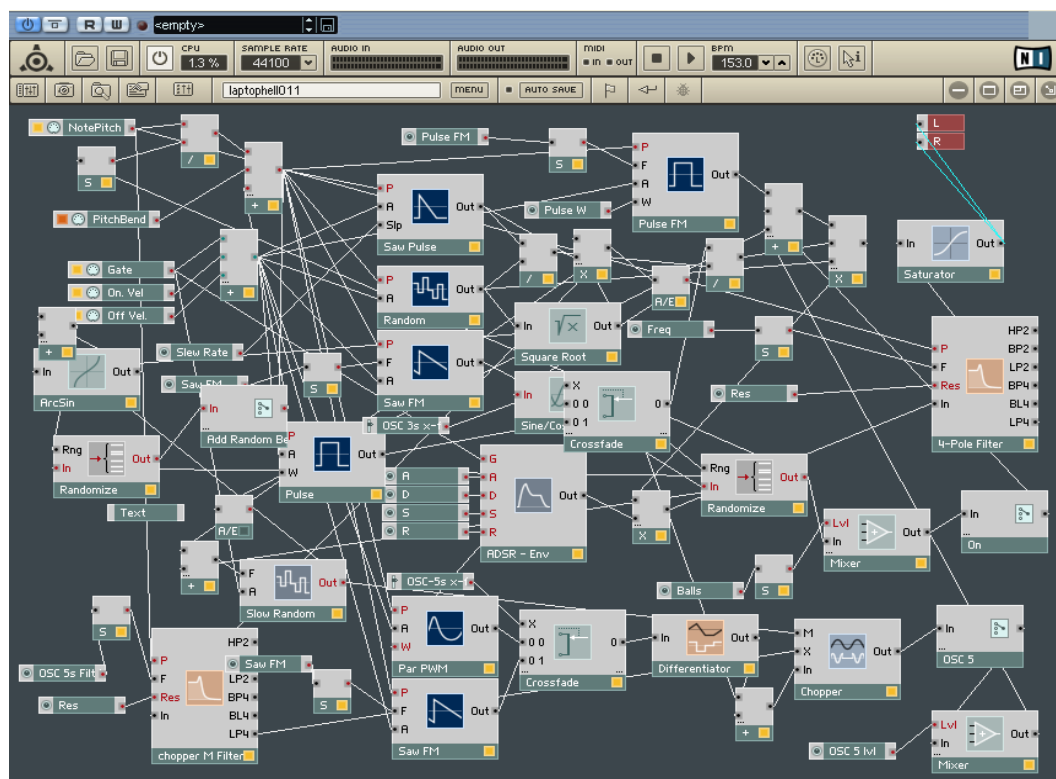


Fig. 17 Exemple entorn *Reaktor*.

4.4.4 SynC Modular

SynC Modular, és un sintetitzador software modular que permet dissenyar sintetitzadors propis per obtenir els sons desitjats. Després de que el seu creador es mogué a una empresa privada per treballar en el desenvolupament de *Reaktor 5*, *SynC Modular* es convertí en programari gratuït.

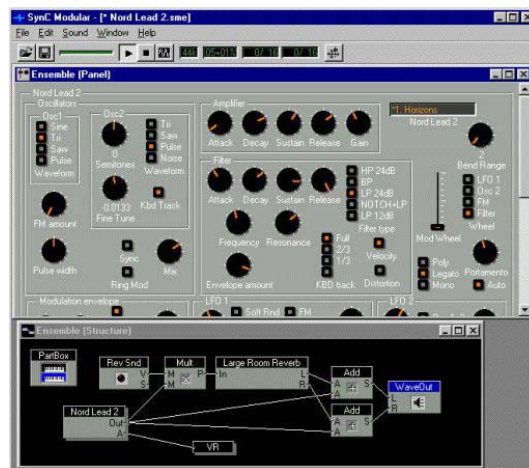


Fig. 18 Programa *SynCModular*.

4.4.5 SynthMaker

SynthMaker, recentment integrat dins del programari propietari *FlowStone*, és una eina de programació d'àudio que permet a qualsevol crear sons propis, instruments virtuals i efectes sense la necessitat d'escriure res de codi. Aquests instruments i efectes poden ser utilitzats per tocar mitjançant MIDI o com a plugin VST (*Virtual Studio Technology*), amb el que poder crear nova música vinculant-se en qualsevol software dels més populars.

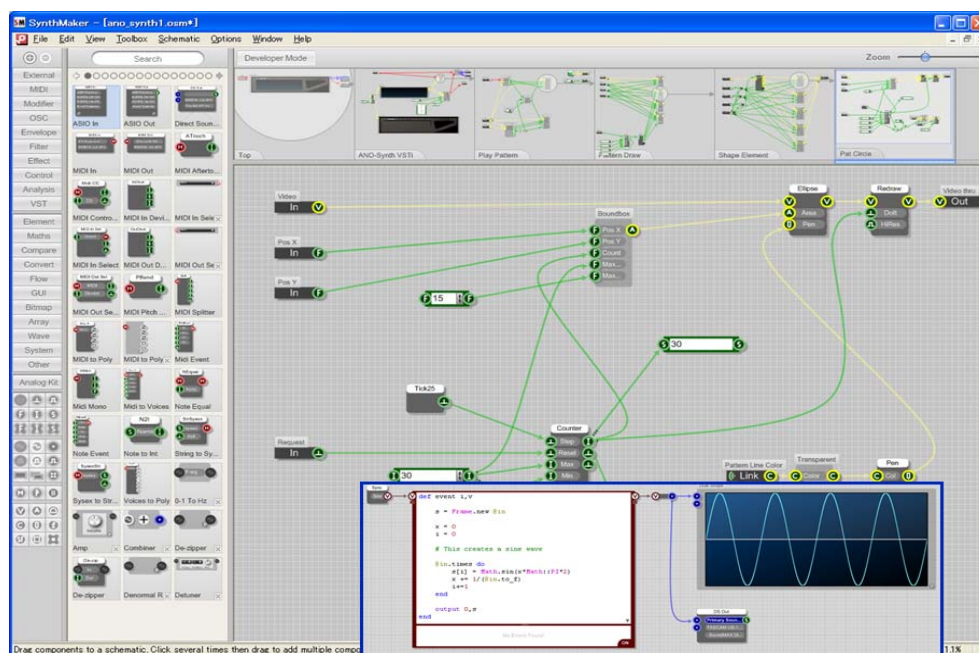


Fig. 19 Exemple entorn *SynthMaker*.

4.5 Editors de So

4.5.1 *SPEAR*

SPEAR, provinent de *Sinusoidal Partial Editing Analysis and Resynthesis*, és una aplicació que com el seu nom indica serveix per a anàlisi, edició i síntesis d'àudio. El punt clau d'aquest programa és en la peculiar presentació gràfica i la capacitat que ofereix per treballar en les representacions sinusoidals separades per harmònics al llarg del temps.

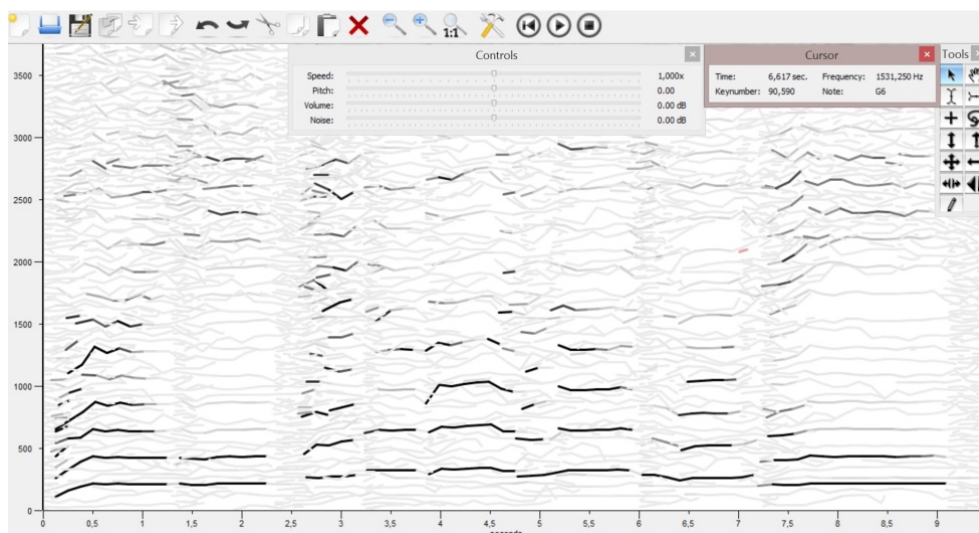


Fig. 20 Interfície d'*SPEAR*.

4.5.2 *Sonic Visualiser*

Sonic Visualiser, té la mateixa funcionalitat que l'anteriorment anomenat *SPEAR* permetent analitzar contingut musical d'arxius d'àudio, tot remarcant-ne la seva interfície d'usuari, que és molt vistosa i completa.

4.5.3 *Altres Editors de So*

Altres alternatives d'editors d'àudio en poden ser *Snd* i *Audacity* entre els softwares lliures o *SoundForge* propietat de *Sony*.

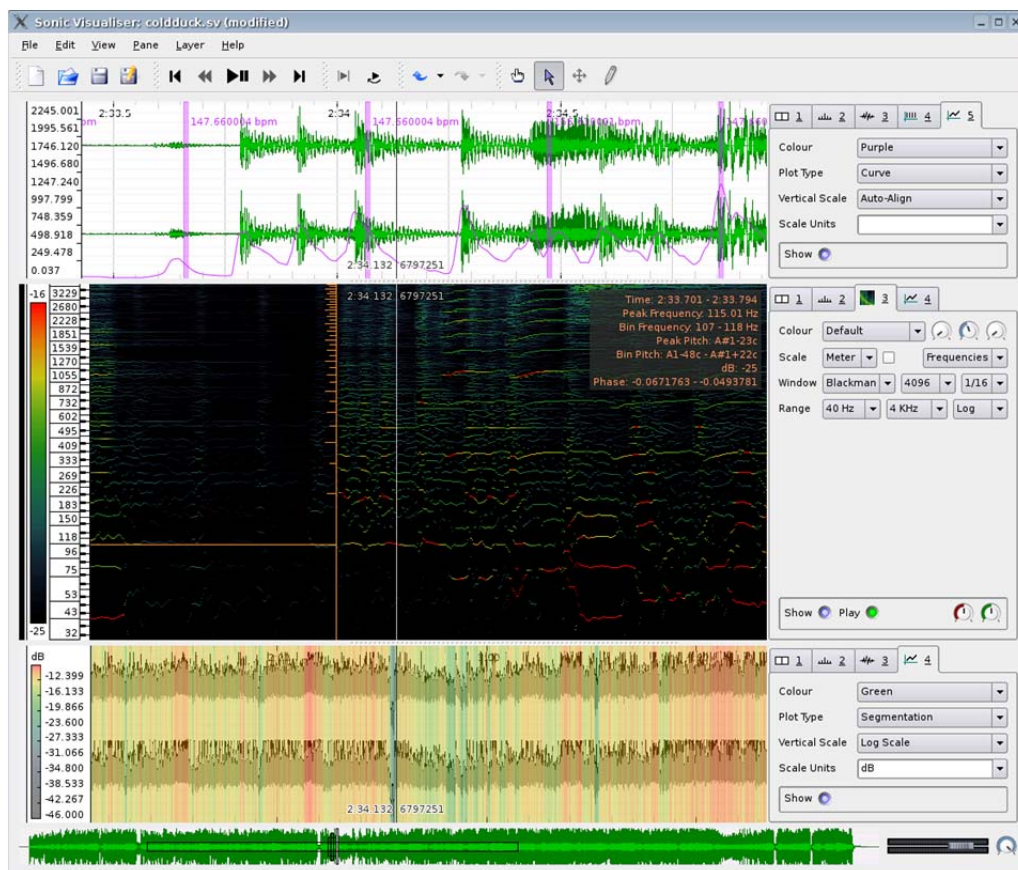


Fig. 21 Interfície de *Sonic Visualiser*.

5 Filtres

Centrant-nos en processament de senyals, un filtre és un dispositiu que modifica un component no desitjat d'un senyal, variant la seva amplitud i la seva fase.

Els filtres es caracteritzen per la seva funció de transferència que de forma matemàtica representa el comportament que tindrà el senyal d'entrada, es a dir els canvis en fase i amplitud que rebrà mitjançant el filtre. Classifiquem els filtres en funció d'aquesta alteració tot distingint els següents tipus:

- **Filtre passa-baix:** deixa passar aquelles freqüències que estan per sota d'una de determinada.
- **Filtre passa-alt:** deixa passar aquelles freqüències que estan per sobre d'una de determinada.
- **Filtre passa-banda:** deixa passar aquelles freqüències situades dins d'un interval concret.
- **Filtre banda eliminada:** deixa passar totes les freqüències a excepció de les compreses dins d'un interval concret.

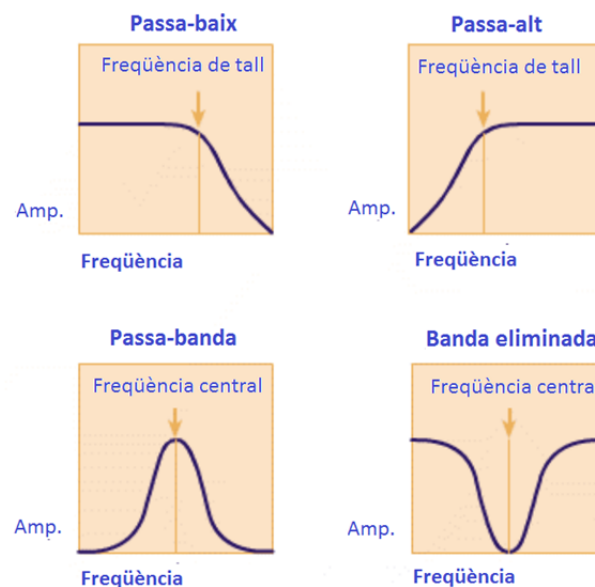


Fig. 22 Tipus de filtres.

Un exemple pràctic on s'utilitzen els filtres anteriorment anomenats és en un equalitzador, que no deixa de ser un configurador d'amplituds per bandes de freqüències sonores. Ja sigui per l'estil musical o per les característiques sonores de l'equip de música, regular el so de sortida tot amplificant o atenuant certes freqüències és essencial per a poder aconseguir la sonoritat personalment desitjada.

Com que els filtres son una peça clau en la modificació del so, s'ha fet un petit estudi de funcions en *MatLab* que poden ser útils per un desenvolupament futur.

A continuació es mostren alguns apartats d'un exemple d'equalitzador de 3 bandes exposat per complet a l'**Annex B** tot mostrant la configuració dels filtres i els resultats obtinguts un cop tractada la senyal.

```
[x, fs] = wavread('Fixer_Orig.wav');
N = 100; % Ordre dels filtres FIR

% Els guanys per cada una de les 3 bandes
gPB = 0.7;
gPM = 5;
gPA = 8;

% Construcció dels filtres
Wn = .09; % Freqüència de tall normalitzada
PB = fir1(N, Wn); % Filtre Passa-Baixos
Wn1 = [.09, .22];
PM = fir1(N, Wn1); % Filtre Passa-Banda
Wn2 = .22;
PA = fir1(N, Wn2, 'high'); % Filtre Passa-Alts
```

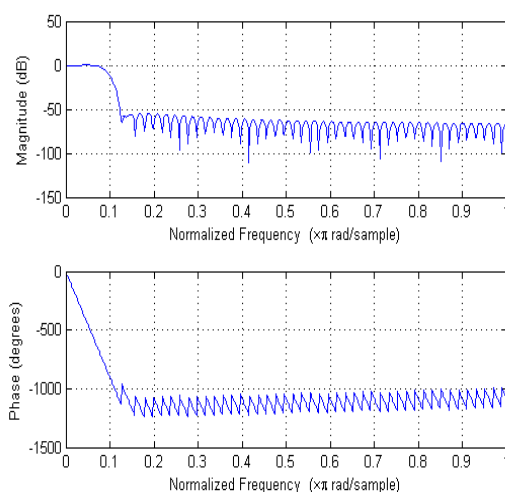


Fig. 23 Filtre Passa-baix.

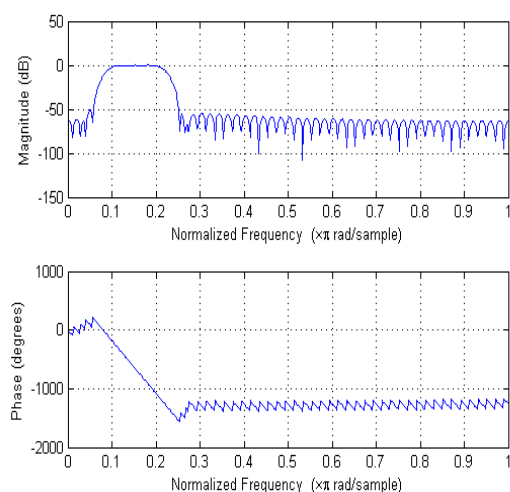


Fig. 24 Filtre Passa banda.

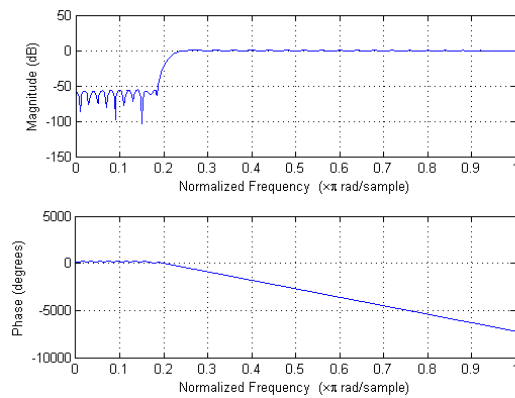


Fig. 25 Filtre Passa-alt.

% Filtrar i graficar els 3 espectres amb el guany aplicat

y1 = conv(PB, x);

y2 = conv(PM, x);

y3 = conv(PA, x);

yA = gPB * y1;

yB = gPM * y2;

yC = gPA * y3;

yD = yA + yB + yC; % Senyal resultant equalitzada

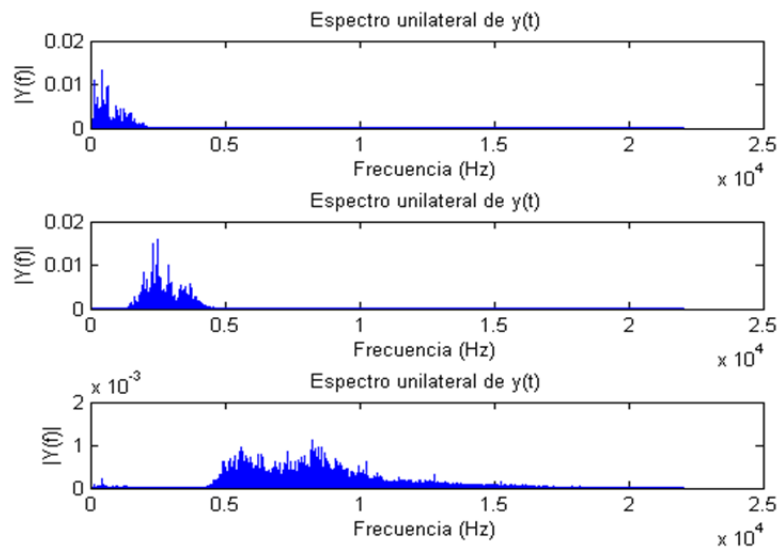


Fig. 26 De dalt a baix, parcial yA, yB i yC amb el guany aplicat.

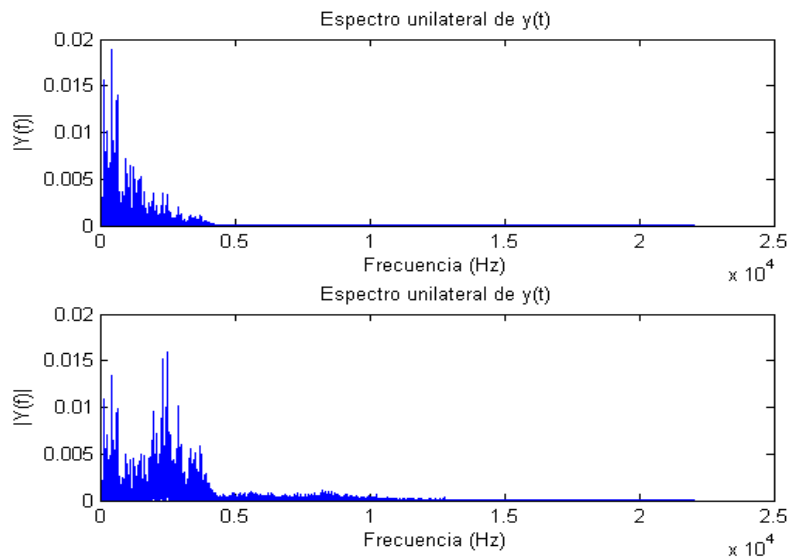


Fig. 27 Gràfica comparació espectre senyal original amb la equalitzada.

Com es pot comprovar en la *figura 27* hi ha una clara diferència entre les freqüències registrades a la senyal d'entrada i la de sortida, on les freqüències baixes s'han vist atenuades lleugerament mentre que les mitges i les altes s'han potenciat.

Hi ha moltes maneres i tecnologies per construir filtres, i es que una mateixa funció de transferència es pot implementar de diferents maneres. A continuació es comenten els més freqüents:

- **Filtres analògics:** basats en una combinació de bobines, resistències i condensadors. S'utilitzen per senyals en temps continu. Poden també incloure amplificadors operacionals o altres elements actius.
- **Filtres digitals:** s'obtenen implementant un algorisme matemàtic corresponent a la funció de transferència desitjada. Aquests s'empren per senyals de naturalesa discreta o contínues mostrejades i mitjançant programaris diversos es poden implementar en maquinari digital o sobre un microprocessador.

Ja sigui mitjançant els telèfons mòbils, els equips de so, tele o radio, etc... la transmissió de senyals és un aspecte clau en la nostra vida quotidiana. Les senyals que rebem viatgen tot travessant diversos medis hostils, veient-se inclús transformades per altres elements electrònics, per això es necessari disposar de mètodes per recuperar parts de senyals desitjades.

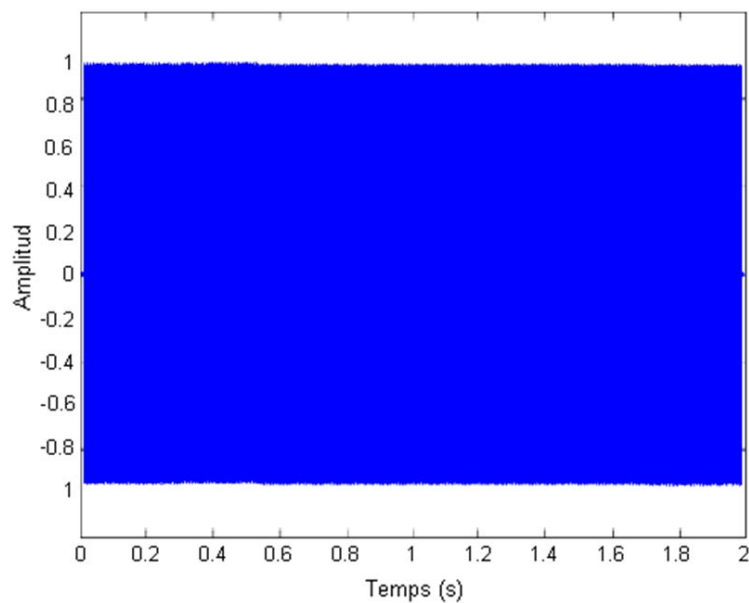
Un altre exemple on s'utilitza un procés de filtratge és al voler aplicar un envoltent determinat sobre un senyal qualsevol.

Aquest envoltent és la forma del senyal en el seu transcurs temporal, tal i com s'ha exposat a l'apartat del timbre, i es pot obtenir d'un senyal a partir de la seva transformada de Hilbert[1].

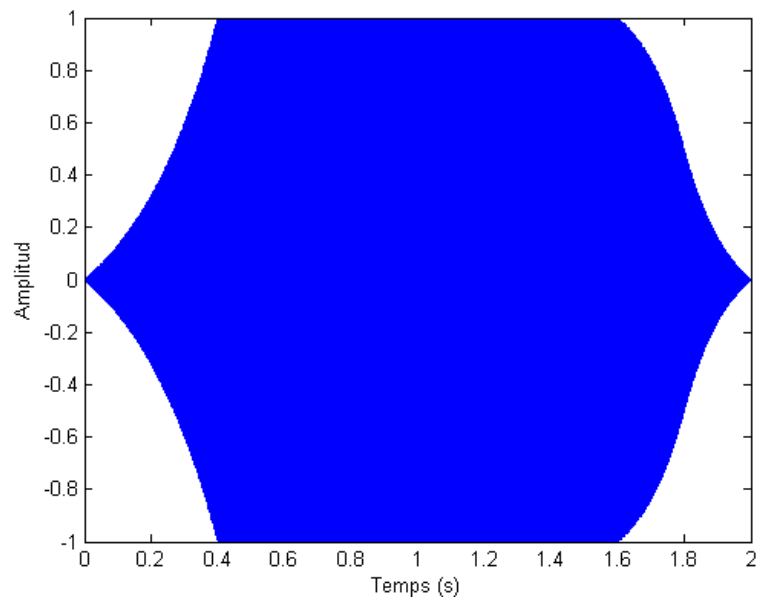
A l'aplicar la transformada de Hilbert es realitza la transformada ràpida de Fourier deixant a zero les freqüències negatives i seguidament la transformada inversa, la part real de la qual és el senyal original, i la part imaginària és la transformada. Amb aquest procés s'obté el senyal analític i per obtenir l'envoltent del senyal original es necessari aplicar el valor absolut en aquest senyal analític.

El següent exemple amb Matlab utilitza la funció *hilbert()* per a extreure l'envoltent d'un senyal i aplicar-li a una altre qualsevol, es a dir filtrar-lo.

```
% Senyal inicial al qual li volem donar forma  
y0
```



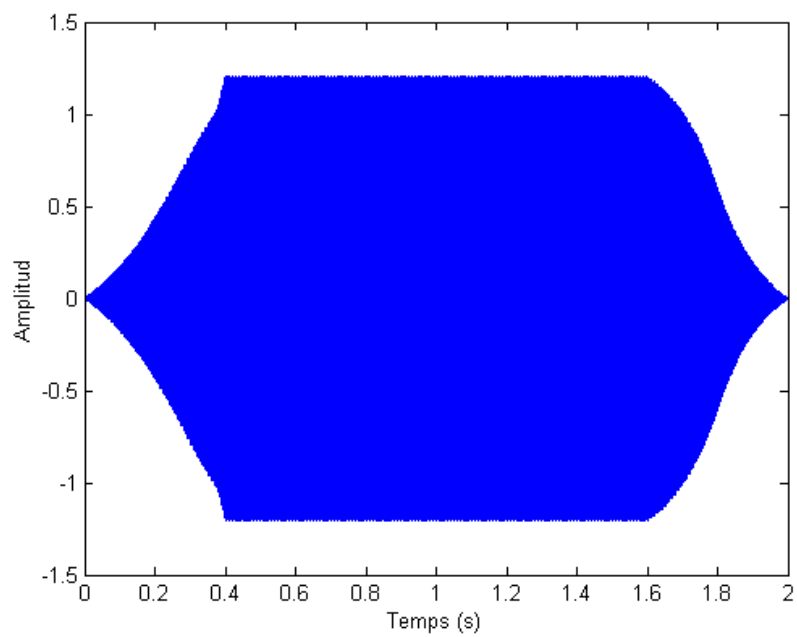
```
% Senyal secundari del qual extraiem l'envolvent
y1
```



```
env = abs(hilbert(y1)); % obtenció de l'envolvent a partir de la transformada de Hilbert

len2 = min([length(y0) length(env)]); % s'igual a la longitud dels vectors
y0 = y0(1:len2);
env = env(1:len2);

Yresult = y0.*env; % es multipliquen els elements un a un per aplicar l'envolvent
```



Si es volgués extreure la transformada de Hilbert en Matlab hauríem d'utilitzar la següent comanda *imag(hilbert())* per extreure'n també la part imaginària.

Amb la funció *[env, ht] = hilbert_envelope(wf,varargin)* afegida a l'**Annex C** es facilita tan l'obtenció de l'envolvent com la de la transformada de Hilbert, i es dona l'opció de no haver d'utilitzar el *signal processing toolbox*, substituint doncs la funció *hilbert()* anteriorment utilitzada a l'exemple, i deixant desglossada la seva implementació.

6 Síntesi

6.1 Introducció

La síntesi del so és un conjunt de tècniques i procediments que es necessiten per obtenir un so de forma artificial. Això s'aconsegueix amb medis electrònics, amb els quals és possible generar senyals elèctriques que vibrin, de la mateixa manera que ho fa qualsevol font sonora. Aquestes senyals o corrents elèctrics variables es poden convertir en so, sempre que les seves freqüències estiguin compreses dins de l'espectre d'àudio, es a dir, dins del rang de freqüències que l'oïda humana pot percebre. Així doncs, la síntesis del so consisteix en treballar amb senyals elèctriques i circuits que les manegin per després convertir-les en so a través d'un amplificador i un altaveu.

El circuit elèctric elemental capaç de generar una corrent variable o vibració elèctrica és l'oscil·lador. Una font de so d'on surt la primera vibració, que esdevindrà després d'un cert procés, la senyal final.

6.2 Mètodes de Síntesi

En general, qualsevol dispositiu capaç de generar una freqüència compresa entre 20 i 20.000 Hz pot produir so. Però el concepte de síntesis va més enllà de la simple generació d'una vibració sonora. En general, s'anomena síntesis a la generació de sons que imiten a d'altres. Aquest és el punt de partida. Però també s'anomena síntesis a l'obtenció de nous sons, com els que podem escoltar en alguns sintetitzadors actuals. Llavors, síntesis de so no és simplement generar so artificialment sinó aplicar una sèrie de tècniques per treballar amb les qualitats del so i així poder obtenir sons més rics i determinats que, en general, imiten el so natural d'un instrument musical.

En la *figura 28* es pot observar la classificació de les diferents tècniques segons Julius O. Smith[2] ampliada amb alguns de més recents.

Mètodes de domini temporal	Models espectrals (domini freqüencial)	Models físics	Mètodes abstractes
Sampling Concrète Granular Scanned Synthesis	Additive Phase Vocoder Sinusoidal (MQ/PARSHL) SMS (Smith & Serra) Subtractive Source-filter (LPC) UPIC (Xenakis) FOF/FOG/PSOLA	Waveguide Finite Element (CORDIS-ANIMA) Karplus-Strong Modal	FM AM Waveshaping Analog simulation (VCO, VCA, VCF) Chaotic systems
Wavetable Particles (Roads) Concatenative Audio Mosaicing (MPEG-7)			
Physically Informed Sonic Modeling (PhISM)			

Fig. 28 Classificació tècniques de síntesis.

A continuació es detallen els mètodes principals i més adients per contextualitzar el concepte síntesi dins de l'anàlisi i de la modificació de la senyal, i així doncs, estudiar si la modificació tímbrica proposada és podria entendre també com un mètode de “re-síntesi”.

6.2.1 Síntesi Additiva

És la tècnica de síntesis de so que crea timbres sumant ones sinusoidals, que és el mateix que afegir parcials harmònics o inharmonics en base a una freqüència fonamental.

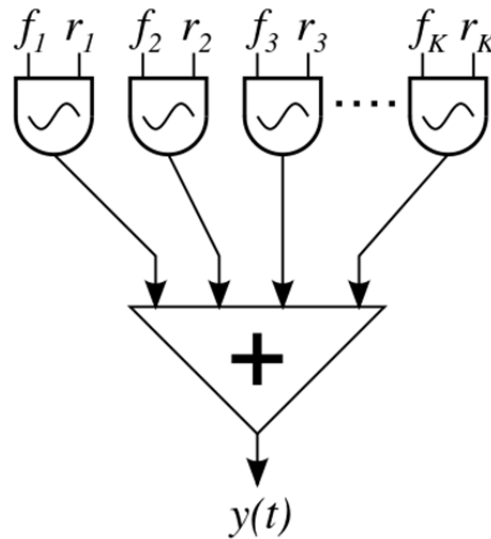


Fig. 29 Diagrama esquemàtic de la síntesi additiva. Les ones sinusoidals representades per freqüència f_k i amplitud r_k .

Aquests components sinusoidals són de major freqüència que la fonamental i s'anomenen sobretons. Els parcials harmònics son múltiples enters de la freqüència fonamental, i inharmònics son múltiples reals.

La teoria de la síntesi additiva es basa exactament en el teorema de Fourier. Com s'ha explicat a l'apartat del timbre, mitjançant l'anàlisi de Fourier s'estableix que totes les senyals representades com una funció matemàtica poden ser compostes per una suma de funcions sinusoidals de varies freqüències. Així doncs per representar un timbre, la síntesi additiva per components harmònics pot ser expressada matemàticament com:

$$y(t) = \sum_{k=1}^K r_k \cos(2\pi k f_0 t + \varphi_k)$$

on $y(t)$ és la sortida, r_k l'amplitud, f_0 la freqüència fonamental de la nota musical en qüestió i φ_k el desfasament, respectivament amb k que és l'harmònic parcial del total de K harmònics.

Com que per una síntesis més dinàmica i realista, el resultat de la suma d'aquests sinus ha de ser variable en funció del temps, es necessita un envoltent que s'encarregui d'ajustar l'amplitud sobre cada parcial i així doncs creï una estructura corresponent pel comportament del so en el temps. Per això es pot modificar la funció anterior i deixar l'amplitud de cada harmònic en funció del temps:

$$y(t) = \sum_{k=1}^K r_k(t) \cos(2\pi k f_0 t + \varphi_k)$$

Tot i que teòricament amb la síntesis additiva tenim la capacitat de generar qualsevol so, i presenta un gran avantatge en vers a les transformacions, no deixa de ser una tècnica poc atractiva pel que fa al control i recursos. Un timbre complex pot requerir 100 o molts més oscil·ladors sinusoidals individuals, cadascun amb la seva pròpia funció d'amplitud i freqüència. Tot i així s'han estudiat mètodes d'optimització en tipus de síntesis de model sinusoidal les quals poden ajudar a reduir aquests oscil·ladors de milers a centenars sense notar diferències perceptibles a l'oïda humana i així doncs minimitzar l'impacte en quan a recursos[3].

Aquest mètode adopta la anàlisi de Fourier, que descriu la representació dels components de freqüència d'un so com a suma d'ones sinusoidals pures. Aquesta anàlisi és duta a terme en estat d'equilibri per donar una aproximació de l'espectre d'aquest so. Com que la majoria dels sons naturals son espectralment dinàmics, no n'hi hauria prou amb un sol anàlisi de Fourier per representar un so. Però “finestrejant”, una Transformada Ràpida de Fourier (FFT) recull varies d'aquestes aproximacions y les encadena per predir millor l'espectre d'un so en el temps.

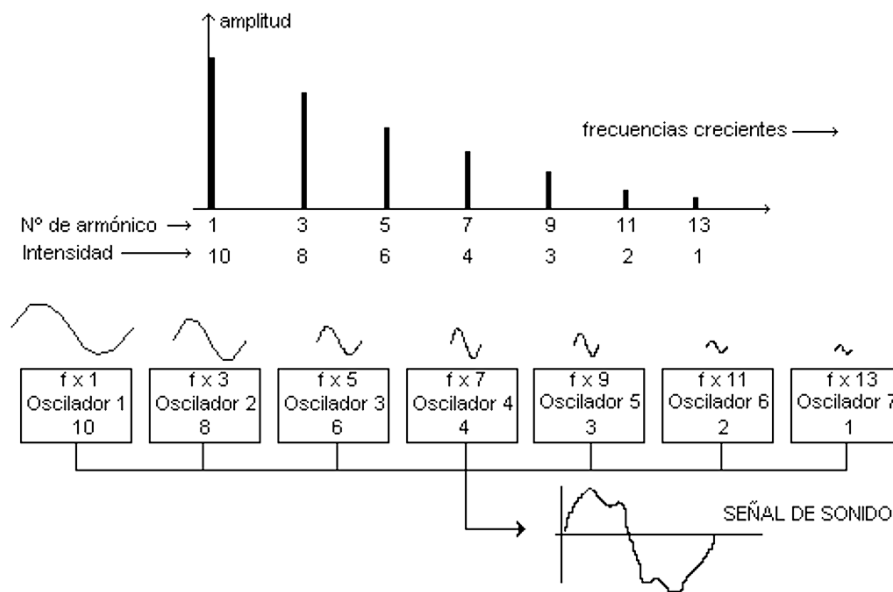


Fig. 30 Diagrama detallat de la síntesi additiva.

Aquest tipus de síntesi el trobem en alguns sintetitzadors moderns, però un altre exemple en podria ser l'orgue, on la suma d'ones sinusoïdals generades per l'aire que passa pels diferents tubs a la vegada, acaba formant el timbre d'una nota.

Per acabar d'entendre millor la síntesi additiva i analitzar el comportament del so a mesura que es van afegint harmònics s'han fet unes proves amb Matlab del que podria ser una síntesi tímbrica amb els 7 principals harmònics del clarinet.

```

fm = 44100; % Freqüència de mostreig
f = 440; % Freqüència fonamental

w = 2*pi*f
d = 2; % duracio en segons
t = 0:1/fm:d; % vector temporal

y = sin(w*t) + 0.75*sin(3*w*t) + 0.5*sin(5*w*t) + 0.14*sin(7*w*t) + 0.5*sin(9*w*t) + 0.12*sin(11*w*t) + 0.17*sin(13*w*t);

```

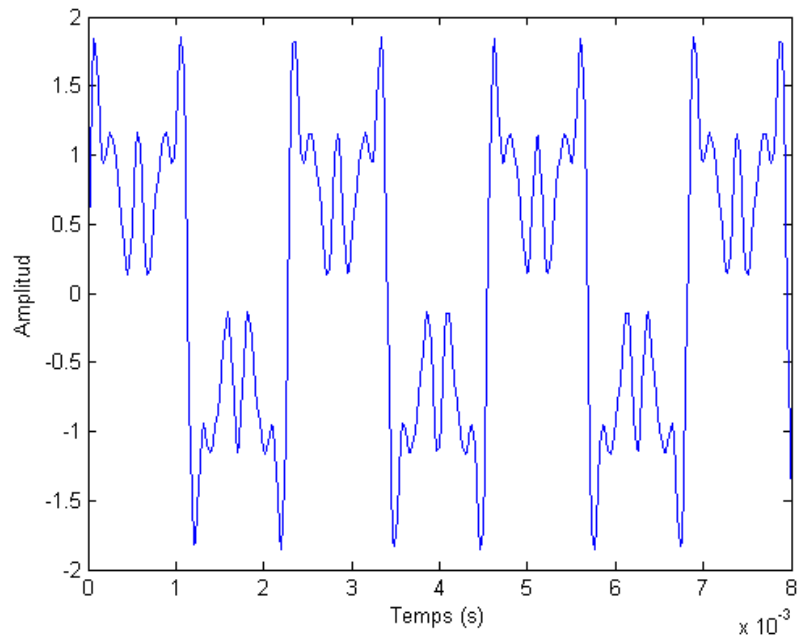



Fig. 31 Forma d'ona resultant.

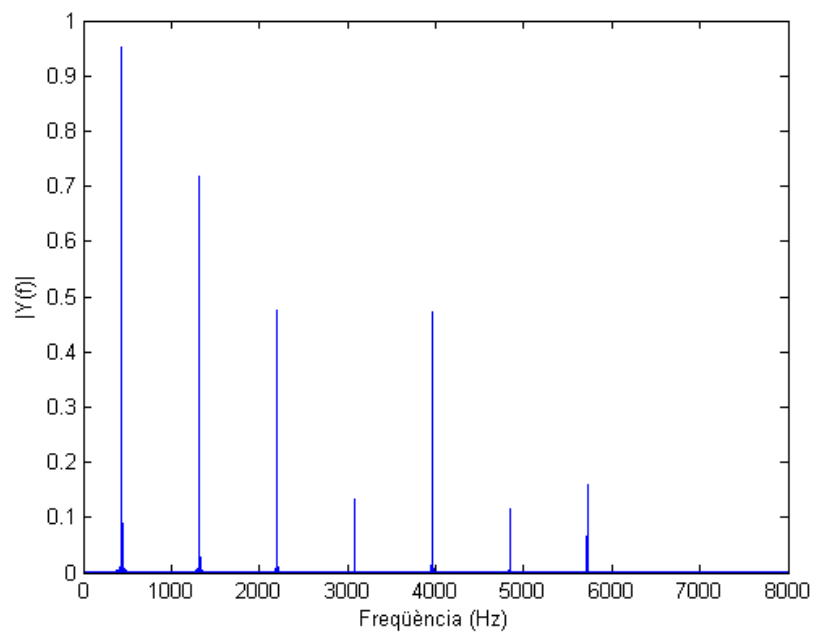


Fig. 32 Espectre unilateral de $y(t)$.

6.2.2 Síntesi Subtractiva

És la tècnica de síntesis de so que s'encarrega de generar formes d'ona complexes tot filtrant freqüències determinades per aconseguir el so desitjat.

En aquest tipus de síntesis els filtres són una peça clau, com millors siguin i més opcions de filtres diferents hi hagi per triar, més adaptat podrà ser el resultat final.

Consisteix en generar una ona amb gran quantitat d'harmònics per després, a partir de filtres, eliminar o controlar la intensitat dels harmònics fins a igualar l'espectre harmònic donat.

Aquest mètode requereix un únic oscil·lador que s'encarrega de generar l'ona de gran riquesa harmònica, tot i així com més harmònics i complexitat és necessitaran més filtres diferents.

Les formes d'ona més utilitzades en els sintetitzadors que es basen en aquest mètode, són la quadrada, la triangular i en especial la de dent de serra, i pel que fa als filtres, el basa-alts, passa-baixos i passa-banda.

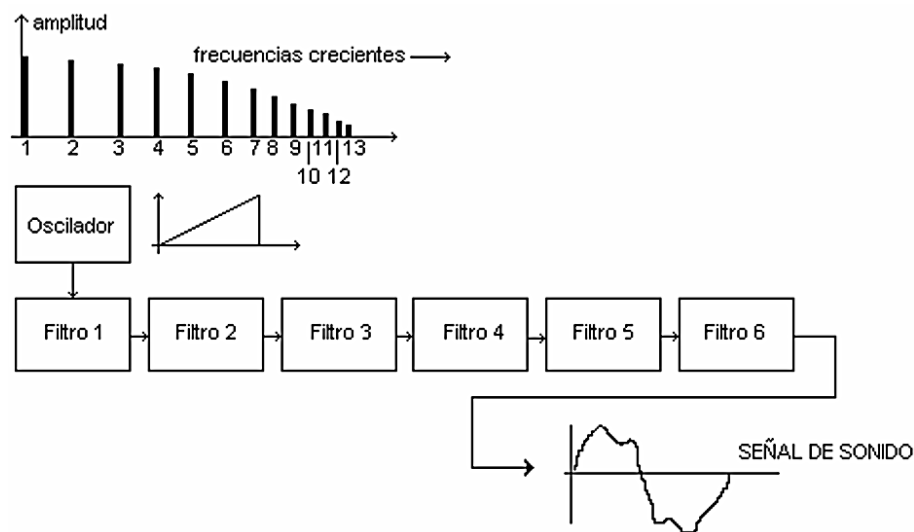


Fig. 33 Diagrama detallat de la síntesi subtractiva.

En aquest esquema de síntesis subtractiva, cada un dels sis filtres modifica l'amplitud d'un o varis harmònics determinats, modificant així l'espectre original, el qual es molt ric en harmònics. El resultat és una forma d'ona determinada, totalment diferent a la inicial.

Els filtres utilitzats poden ser qualsevol dels exposats a l'apartat de filtres d'aquest treball: passa-alt, passa-baix, passa-banda o banda eliminada.

Degut a la facilitat per a crear sons nous, el mètode de síntesis subtractiva és el més utilitzat en la majoria de sintetitzadors analògics dels 70.

6.2.3 Síntesi per Modulació de Freqüències

Aquesta mètode de síntesi utilitza la mateixa tècnica que la radiotransmissió que tots coneixem, la qual consta d'una ona per augmentar o disminuir la freqüència d'una altra, es a dir, modular-la[4].

L'ona base que encara no s'ha modificat, s'anomena portadora, la que modifica la portadora s'anomena moduladora, la qual no s'escolta, i l'ona resultant s'anomena modulada. Aquesta parella d'ones inicials, que és el mínim necessari per tal de portar aquest tipus de síntesi en pràctica, s'anomenen operadors.

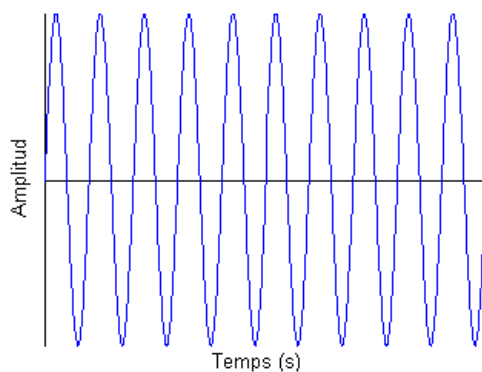


Fig. 34 Ona portadora, o “carrier”.

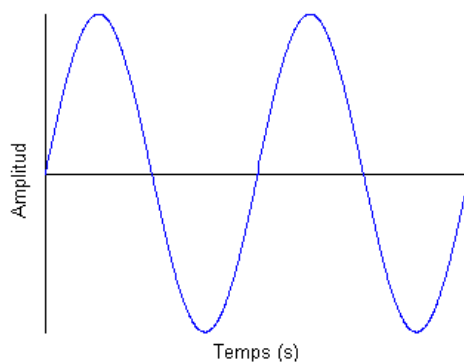


Fig. 35 Ona moduladora.

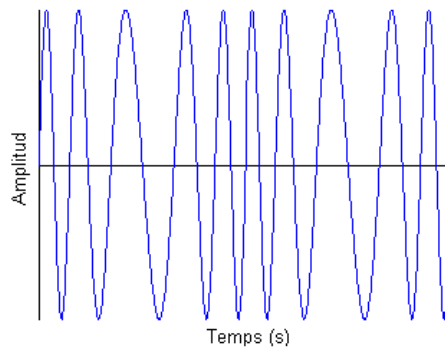


Fig. 36 Ona modulada.

A primera impressió, ens pot estranyar que modulant una freqüència es puguin crear nous timbres, però es cert. Si es modula la freqüència d'un to de manera lenta podem percebre com un efecte *vibrato*, com ara el de l'oscil·lació de veu d'un cantant d'òpera, però en aquests casos de síntesis FM estem parlant de modulacions de 440Hz es a dir fer pujar i baixar el to 440 vegades en un segon, la qual cosa segons les funcions de Bessel comporta l'aparició de noves bandes laterals o harmònics[4].

La freqüència instantània de l'ona portadora és variada d'acord la moduladora i tal quantitat de variació va en funció a la mitjana que es coneix com la desviació de pics de freqüència entre ambdues ones, que a l'hora és proporcional amb l'amplitud de l'ona moduladora.

Els paràmetres d'una senyal amb característiques de freqüència modulada son els següents:

- ***c***, Freqüència de la portadora o freqüència mitjana.
- ***m***, Freqüència moduladora.
- ***d***, Desviació de pics de freqüència.

L'equació per una ona resultat de freqüència modulada amb un pic d'amplitud ***A*** i on ambdós components d'ona són sinusoidals és la següent:

$$e = A \sin (\alpha t + I \sin \beta t)$$

on ***e*** és l'amplitud instantània de l'ona portadora que ja ha sigut modulada, ***α*** és la freqüència de l'ona portadora en ***rad/seg*** i ***β*** és la freqüència de l'ona moduladora també en ***rad/seg***.

I = d/m, és l'índex de modulació o en altres paraules, el rati (divisió) entre la desviació de pics de freqüència i la freqüència moduladora. És obvi que quan ***I = 0*** la desviació pic de freqüència també és ***d = 0*** i per tant no hi ha cap tipus de modulació. Quan ***I > 0***, apareixen altres freqüències o components laterals per damunt i per sota de la freqüència portadora en intervals

proporcionals a la freqüència moduladora. Aquests passen a ser part de l'espectre resultant de la portadora ja modulada.

El numero de freqüències laterals que sorgeixen en FM està relaciona al índex de modulació I de tal manera que a l'incrementar I de 0 en endavant, una quantitat proporcional a l'energia de la portadora es distribueix a cada una de les bandes o freqüències laterals. Al incrementar-se el valor del índex de modulació també s'amplia l'ample de banda de l'ona resultant, i es van produint bandes laterals. Les freqüències del costat de dalt i de sota son intervals proporcionals a la freqüència moduladora m y són simètrics al voltant de la portadora c . Les amplituds de la portadora i les components de les seves bandes laterals estan determinades per funcions de Bessel del primer tipus i ordre- n , així: $J_n(I)$, on l'argument d'aquesta funció es l'índex de modulació I .

La riquesa de la freqüència modulada es basa en que hi ha proporcions, entre les freqüències portadores i moduladores a més a més de valors de l'índex de modulació que produeixen components laterals que poden quedar en un domini de freqüències negatives del espectre. Aquests components negatius es reflecteixen o es desdoblen al voltant de $0Hz$. i es barregen amb els components del domini positiu. La varietat de relacions de freqüències que resulten d'aquesta barreja és d'una gran quantitat de combinacions que a sobre inclouen espectres harmònics i altres d'inarmònics.

Seguint amb el referent del timbre de clarinet, a continuació s'ha exemplificat amb *MatLab* aquest tipus de síntesis per observar i estudiar el seu comportament i comparar-lo amb altres.

```
function y = clarinet_fm(f0, dur, fsamp)
% CLARINET_FM produeix una nota tocada sintetica de clarinet
%
%   us: yy = clarinete(f0, dur, fsamp)
%
%   on:  f0 = freqüència fonamental de la nota
%        dur = duració de la nota
%        fsamp = freqüència de mostreig

te = 0:(1/fsamp):dur;      % eix del temps
fp = f0;                  % portadora
fm = f0;                  % moduladora

% Pel càlcul d'amplitud
[y1,y2] = env_vent(atac, sosteniment, relax, fsamp);

lt = -2*y2+4; % ajust de l'envolvent de l'índex de modulació

faux = 2*pi*fp*te + lt.*cos(2*pi*fm*te);

% Càlcul de la senyal FM
y = y1.*cos(faux);
```

```

function [y1,y2] = env_vent(a,s,r,Fs)
% ENV_VENT genera la funció d'envolvent de l'amplitud i l'índex de modulació
% per a síntesis d'instruments de vent
%
% us: [y1,y2] = woodwenv(a,s,r,Fs);
% on a = temps d'atac
% s = temps manteniment
% r = temps de relaxació
% Fs = Freqüència de mostreig(Hz)
% retorna:
% y1 = envolvent d'amplitut
% y2 = envolvent de l'índex de modulació
%
% nota: l'atac és exponencial, el manteniment constant, i relaxació exponencial

```

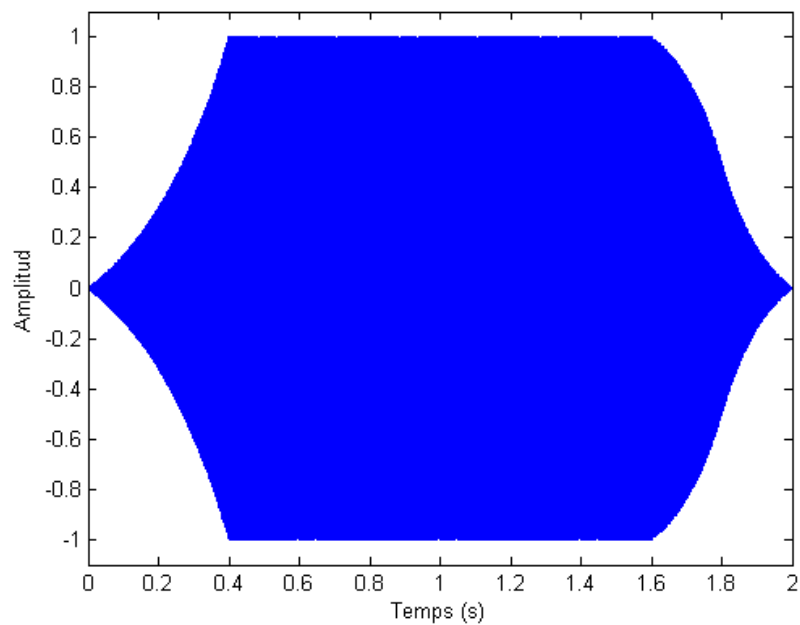


Fig. 37 Ona completa amb forma d'amplitud determinada per l'envolvent.

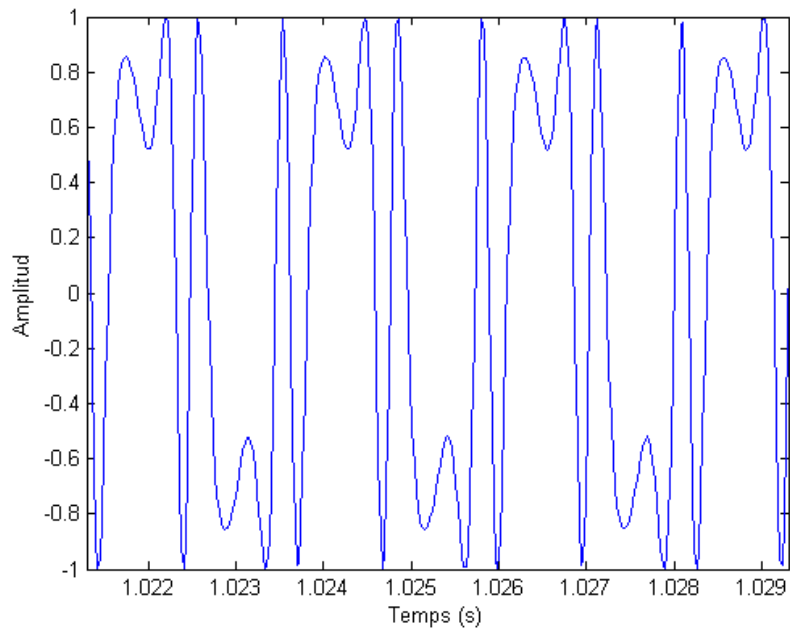


Fig. 38 Ampliació de forma d'ona resultant.

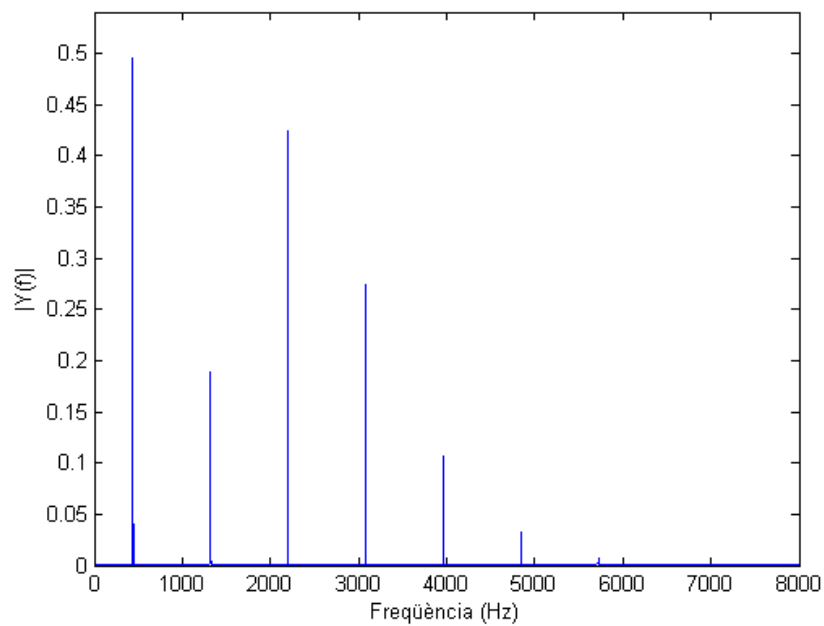


Fig. 39 Espectre unilateral de $y(t)$.

A més a més cal afegir que a la síntesis per Freqüència Modulada s'utilitzen varies d'aquests operadors amb tot tipus de combinacions, es a dir dos moduladors que actuïn paral·lelament sobre una portadora alhora, o be una moduladora sobre un altra moduladora que aquesta sí que actua sobre la portadora, o inclús fer síntesis additiva on sumem varies portadores que cada una te les seves combinacions de moduladores alterant-les....

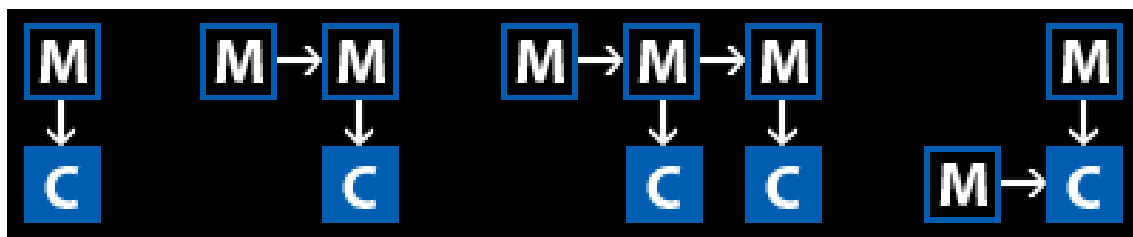


Fig. 40 Exemples de diferents configuracions entre operadors de síntesi FM.

Dins del grup de tècniques de síntesis per modulació, també hi trobem la síntesis per Fase Modulada, que funciona igual però en aquest cas en comptes de canviar la freqüència, l'ona moduladora serveix per canviar l'angle de la fase.

6.2.4 Síntesi Granular

És un mètode de síntesis que es basa en crear so a partir de petites explosions sonores molt curtes, encapsulades en un envoltent i organitzades de manera diferent entre elles, ja sigui de forma seguida o solapada.

Aquestes partícules sonores, base de la de síntesis granular, s'anomenen grànul o quant sonor, i són fragments de so molt curts, la longitud dels quals oscil·la entre 5 i 100 mil·lisegons, d'aquesta manera s'evita que l'oient pugui apreciar un grànul a nivell individual.

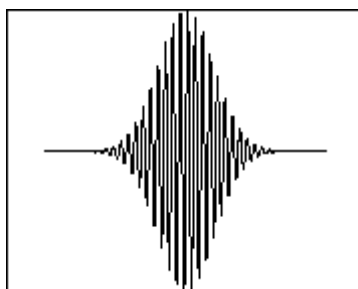


Fig. 41 Grànul sonor sinusoidal.

Els dos mètodes d'organització, o distribució temporal són el mètode sincrònic, on els grànuls són disparats a freqüències regulars per produir sons amb període d'altura particular, i el mètode asíncron, que genera aleatòriament les distàncies entre els grànuls per produir un núvol sonor.

La forma d'ona d'aquests grànuls, inclòs l'envoltent, es defineix per una finestra que pot adoptar diverses formes, i en funció d'aquestes s'acaba definint la informació espectral, i així doncs la composició harmònica.

El processament necessari depèn de la quantitat de dades de control en la generació de grànuls, i com que el numero de paràmetres necessaris per segon es molt gran, es requereix d'una organització d'alt nivell per definir el comportament de tals paràmetres.

6.2.5 *Síntesis Basades en Samples*

Per una banda tenim la tècnica de *Samplejar* (mostrejar), que ni tan sols es pot considerar una tècnica de síntesis de so, ja que simplement consisteix en tenir els sons pre-gravats en memòria per a després poder-los reproduir a diferents velocitats i d'aquesta manera obtenir el so escalat a diferents freqüències, es a dir per a les diferents notes. Els sintetitzadors que utilitzen aquest sistema van esdevenir molt populars sobretot per que també tenien un micròfon i d'aquesta manera cadascú es podia fer els seus propis samples amb els que després crear la musica. Últimament, gràcies a les grans possibilitats d'emmagatzemament, és fàcil trobar llibreries de més de 50 Giga Bytes plenes de samples pre-gravats per a cobrir tot tipus de sons emesos per un mateix instrument, com per exemple un banc de sons de violí, que podria contenir samples amb notes lligades, marcades, amb *staccato*, pinçades, tremolo, i un llarg etcètera.



Fig. 42 Sampler.

Però per altra banda existeixen altres tècniques de síntesis també basades en samples, que utilitzen porcions del so per a després encadenar-les i formar nous sons, com ara les taules d'ona o la síntesis granular i síntesis vectorial.

6.2.6 Taula d'Ones

És la tècnica de síntesis que utilitza les taules d'ona, o wavetables en anglès, que és un espai de memòria on s'emmagatzemen digitalment formes d'ona de diferents instruments, ja siguin gravats, o sintetitzats, per després utilitzar-les per la generació de so.

En aquestes taules normalment n'hi ha prou en guardar un cicle d'ona, i és aquesta una de les característiques més destacades d'aquest mètode de síntesis, ja que no es necessita tan espai d'emmagatzematge com pot semblar. L'altra és que no cal processar el timbre de cada so amb els seus harmònics, sinó que aquest ja ve definit amb la mateixa tros de forma d'ona, i d'aquesta manera s'estalvia molta potència de processament.

Tot i guardar cicles curts d'ona, per obtenir l'alta fidelitat en concepte sonor es necessita una bona qualitat en l'ona origen, i això implica més quantitat d'espai de memòria, per això s'acostumen a utilitzar tècniques de compressió de dades.

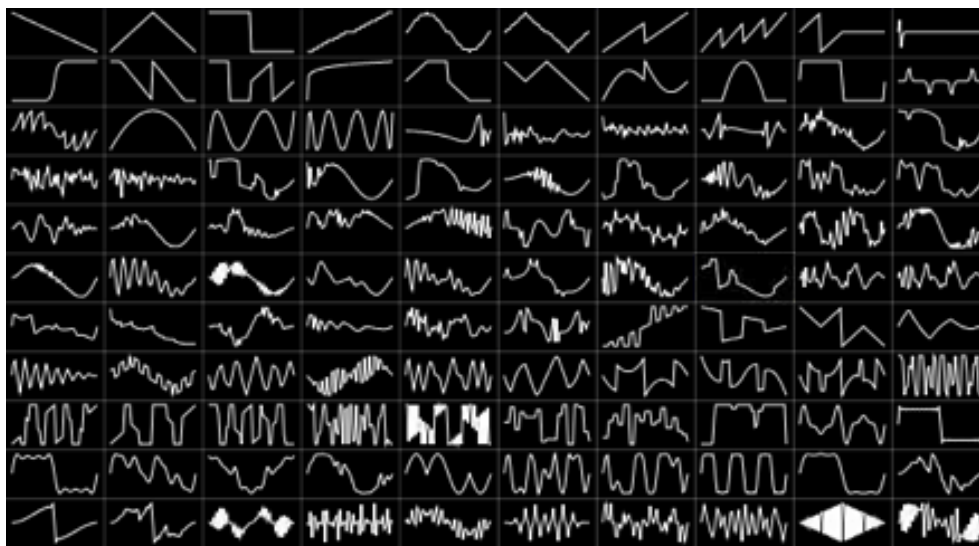


Fig. 43 Representació visual d'una taula d'ones amb l'emmagatzematge de les diferents formes.

Tipus de tècniques dins de la síntesis *wavetable*:

- Generació en bucle o looping. És la principal tècnica emprada en aquest tipus de síntesis, i es basa en mostres de so pre-gravades que son reproduïdes multitud de vegades seguides per construir la senyal original. Per això es divideixen i s'emmagatzemen les dos parts principals de la majoria de sons produïts per instruments: la secció d'atac, que es reproduïx sencera, i la de manteniment, que es reproduïx en bucle cada vegada amb menor intensitat seguint un patró preestablert. Pel que fa a la part de conclusió del so, es regula a través dels filtres d'envolvent, amb l'atenuació de la mateixa secció de manteniment.

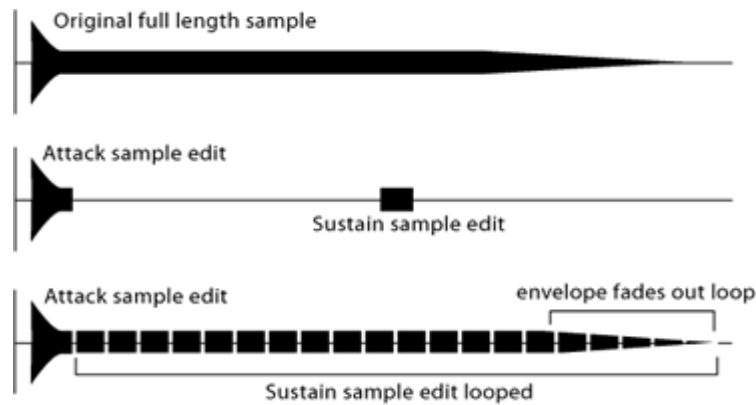


Fig. 44 Esquema de la generació en bucle o looping.

- Desplaçaments de to. És la tècnica utilitzada per poder generar varies notes d'un mateix instrument a partir d'una sola forma d'ona. Això s'aconsegueix augmentant o disminuint la freqüència o aplicant varis filtres a la senyal i la seva finalitat és la de reduir molt l'espai d'emmagatzematge.
- Interpolació. És una tècnica matemàtica necessària per reconstruir millor la senyal original quan aquesta és gravada en freqüències baixes per tal de reduir espai en memòria.
- Sobremostreig. És una tècnica utilitzada només en alguns sons particulars impossibles d'encadenar en forma de bucle, o difícils de convertir en diferents tonalitats, i contràriament a la interpolació, es basa en l'emmagatzematge d'una forma d'ona per cada freqüència, així doncs necessitant molt més espai en memòria.

Així doncs les principals diferències que existeixen entre la síntesis per taula d'ona amb la síntesis basada en *samples* és que la primera sempre està generant en temps real la forma d'ona a mesura que la CPU processa la seqüència i l'altra és que per taula d'ones només cal emmagatzemar un o dos períodes de longitud com a molt.

6.2.7 Síntesi per Modelatge Físic

És la síntesi de so a partir d'un conjunt d'equacions i algorismes que simulen un instrument real, o qualsevol altra font sonora i física. Això implica crear models dels components d'aquests instruments musicals, i crear sistemes que defineixin l'acció, els filtres, els envoltants i altres paràmetres que afecten a l'espai temporal de la senyal. La definició que se li pot donar d'aquesta manera a un instrument és il·limitada, ja que es pot combinar qualsevol model donat amb modulacions de to, freqüència i contorn. Per exemple, es podria representar el model d'un violí al qual se li aplica una distorsió de guitarra elèctrica, o el model de l'acció d'un martell de piano, que és la peça que pica a les cordes d'aquest iniciant la vibració i so de l'instrument. És quan el conjunt de paràmetres inicials s'executen a través de la simulació física que es genera el so.

Tot i que el modelatge físic no era un nou concepte, no va ser fins el desenvolupament de l'algorisme Karplus-Strong i l'augment de la potencia en els sistemes de processament de senyal digital que les implementacions comercials fossin factibles, i van evolucionar encara més amb el major processament dels ordinadors.

Els components del model físic en instruments de vent es poden simplificar en tres. Consten d'una relació lineal entre la impedància de pressió acústica i el flux a l'embocadura de l'instrument, un oscil·lador que modela la forma de vibració de la canya, i un acoblament de característiques no lineals entre la pressió d'aire aplicat a l'instrument i la col·locació de la canya a l'embocadura[5].

A la figura 45 [6] es mostra un exemplar de diagrama de síntesis per model físic de tipus waveguide d'un instrument de vent fusta cilíndric i de canyeta única, com el clarinet. D'esquerra a dreta es divideix el diagrama en tres parts: la part de la canyeta, dels forats i de la campana.

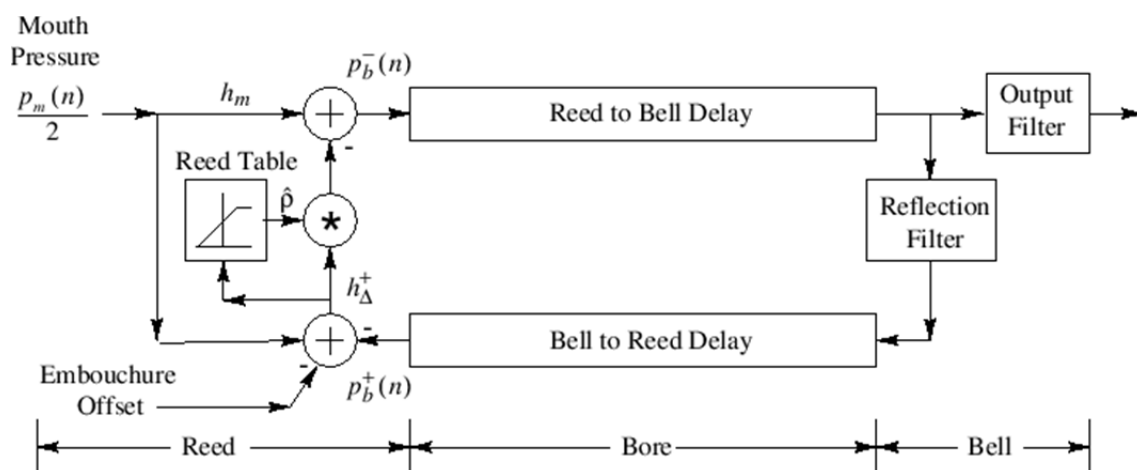


Fig. 45 Esquema waveguide per a instrument similars al clarinet.

A continuació es mostren alguns dels paràmetres i els càlculs utilitzats en un exemple en Matlab de síntesi per modelatge també d'un clarinet[7]. És un exemple simplificat que utilitza un flux estàtic, es a di sense utilitzar equacions diferencials, simplificant també la part de simulació de la canyeta sense tenir en compte la seva massa, i també s'obvia el comportament que podria afegir l'última peça de l'instrument, la campana.

```

% costants mol·leculars
c = 347; % velocitat del so, m/s
rho = 1.2; % densitat de l'aire, kg/m^3

% paràmetres del tub
l = .48; % llargada del tub, m
a = 0.01; % radi del tub, m
S = pi*a*a; % secció de creuament del tub, m^2
Z0 = rho*c/(pi*a*a); % característiques de l'impedància

% variables del modelatge waveguide; N, upper i lower son paràmetres per al filtratge amb retard
y0 = zeros(soundSamples, 1); % buffer de sortida per l'embocadura
yL = zeros(soundSamples, 1); % buffer de sortida per al final del tub

% elements de filtratge en waveguide
bTL = aRL + bRL % vectors parametritzats de reflexió i transmissió d'un tub cilíndric de final obert

stateRL = zeros(max(length(bRL), length(aRL))-1, 1);
stateTL = zeros(max(length(bTL), length(aRL))-1, 1);

% modelat "ONE-WAV propagation loss" parametrítza bL i aL
lambda = freqz(bL, aL, 'whole', fftBandwidth);

R0 = .9; % reflexió a l'embocadura

% paràmetres físics de la canyeta
aw = 0.015; % amplada efectiva per al flux d'entrada, m
S = 0.034 * aw; % superfície canyeta, m^2
k = S * 10000000.0; % duresa de la canyeta, N/m
H0 = 0.0007; % obertura/posicionament canyeta, m (0.4-1mm)

% pressió de l'aire a través de la boca, Pa
Ndecay = .2*fs;
Nattack = .1*fs;
pm = [0; 5000* linspace(0, 1, Nattack)'];

U = zeros(soundSamples, 1); % buffer de volum de flux

for n = 1: soundSamples

    % càlcul de la sortida del model de la canyeta
    dp = abs(pm(n) - y0(n-1)); % pressió delta de la canyeta
    x = min(H0, dp*S/k); % desplaçament de la canyeta (sense paràmetres de massa)
    U(n) = aw*(H0-x)*sqrt(dp*2/rho); % volum del flux admes per la canyeta
    pr = U(n)*Z0;

    % obté les senyals amb retard
    [outL, upperLambda] = filter(bL, aL, upper(ptr), upperLambda);
    [out0, lowerLambda] = filter(bL, aL, lower(ptr), lowerLambda);

    % escriu a les senyals amb el retard
    upper(ptr) = pr + R0*out0;
    [lower(ptr), stateRL] = filter(bRL, aRL, outL, stateRL);

    % buffers de sortida
    y0(n) = out0 + upper(ptr);
    [yL(n), stateTL] = filter(bTL, aTL, outL, stateTL);

end

```

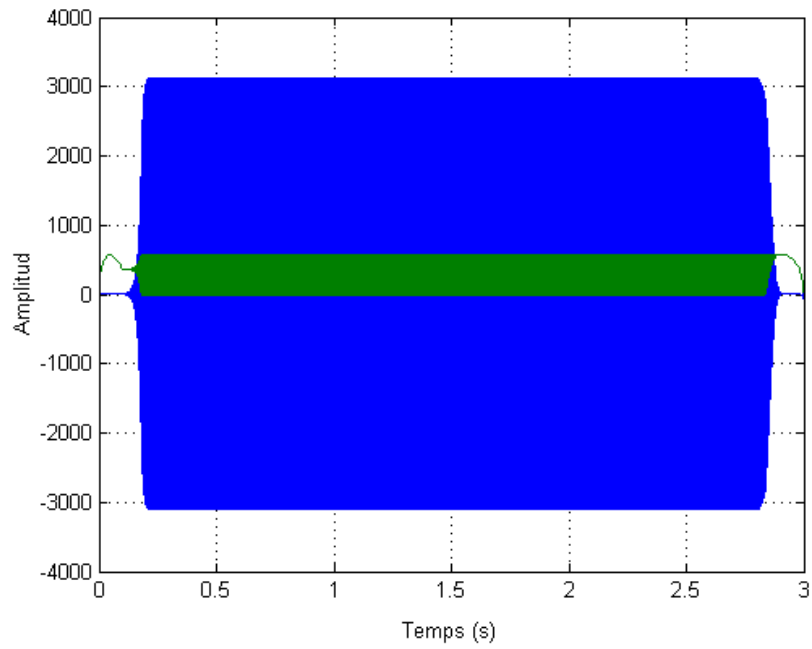


Fig. 46 Senyal de sortida sonora de 3 segons de duració on es veu la relació entre l'amplitud en db. del flux d'entrada en el que es basa el procés (verd), i la resultant (blau).

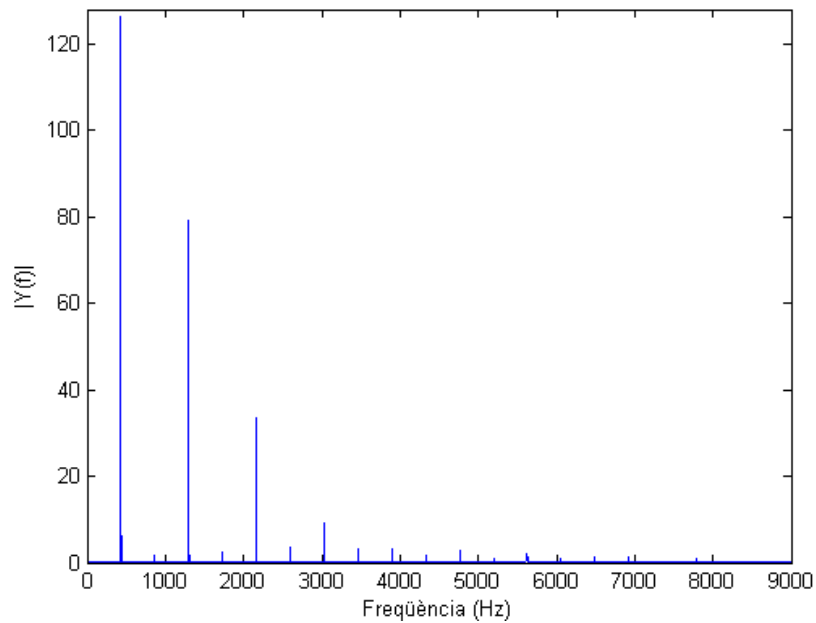


Fig. 47 Espectre unilateral de $y(t)$ amb freqüència fonamental 433 Hz.

Modificant el paràmetre de longitud de tub, es a dir l'obertura del tub donada per la digitació dels dits de l'interpret, i analitzant els components harmònics resultants de la síntesi es pot observar com hi ha un canvi respecte als resultats de l'espectre anterior. El fet de que hi hagin aquestes diferències és un indicador de que aquesta síntesi es pot atansar molt més a la veritable

funció de transferència sonora que pot representar un instrument, la majoria dels quals, tenen formes espectrals ben diferents en funció de la nota interpretada.

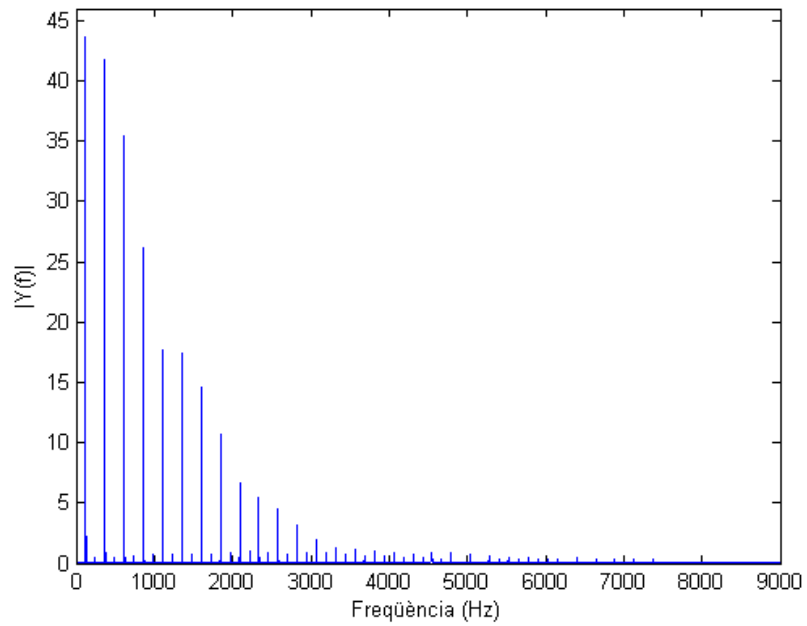


Fig. 48 Espectre unilateral de $y(t)$ amb freqüència fonamental 123 Hz.

7 Disseny

7.1 Introducció, Acotant el Sistema

Va ser a finals dels anys setanta i amb la possibilitat de la computadora com eina per anàlisis i síntesis de so, quan Jean Claude Risset amb un grup de treball als laboratoris Bell d'Estats Units aconseguiren categoritzar i sintetitzar varis sons del món físic real. Aquest treball d'experimentació aconseguí resoldre l'incògnit que prova la importància del desenvolupament espectral a tots els sons que s'originen amb medis físics com un instrument musical real.

Risset va veure que en aquests sons les intensitats a les seves components harmòniques són dinàmiques i variables. L'energia en aquestes components moltes vegades es desenvolupa de forma complicada, particularment al inici durant l'atac del so i al final amb el seu esvaïment, però la part central sí que normalment és constant i periòdica[8].

És per aquest mateix motiu que de moment es decideix fer un sistema que es basa en la part més estable de les notes musicals, la part del sosteniment, on els components espectrals permetran un millor maneig gràcies a la seva periodicitat.

Es fa una suposició teòrica de que el senyal d'entrada serà “pla”, acotant el timbre com a part harmònicament estable, i desestimant els envoltants i variacions d'amplitud.

Cal aclarir, però, que el sistema ha de permetre l'entrada del timbre original en qualsevol de les notes de l'escala, es a dir, permetre l'entrada de diferents configuracions harmòniques, però com s'ha explicat abans, se'n desestimarà les alteracions d'aquestes en el transcurs temporal de cada nota, altrament anomenat envoltent.

També es decideix fixar un timbre inicial d'entrada, el de l'instrument que s'interpreta, en aquest cas el clarinet, i acotant els possibles timbres resultants als que tinguin comportaments d'envoltent i característiques espectrals similars al clarinet, com per exemple un saxo, o un oboè. D'aquesta manera començar amb un objectiu més factible per després valorar fins a quin punt es poden aplicar procediments similars per a altres instruments.

Un cop establertes aquestes normes, es du a terme l'anàlisi del so del clarinet.

7.2 Primer Disseny: Anàlisi del Clarinet i Suma Freqüencial

Un petit estudi del clarinet pot ser analitzar el seu rang freqüencial, i per fer això ens basarem en la seva tessitura, que és precisament el conjunt de notes les quals l'instrument és capaç d'emetre, des de la més greu fins a la més aguda.

En el cas del clarinet, podríem representar la seva tessitura sobre el pentagrama tal i com mostra la *figura 49*.

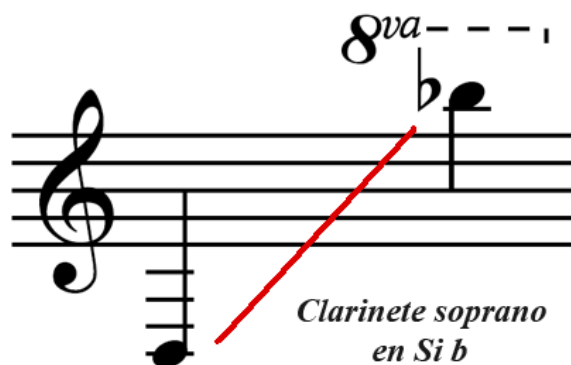


Fig. 49 Tessitura clarinet.

Un cop es coneixen el seu rang en nomenclatura musical, podem extreure'n la seva equivalència en freqüències fonamentals a partir de la taula de l'**Annex A**.

S'ha implementat un script en MatLab per intentar modelitzar un clarinet a partir de la suma de totes aquestes freqüències, però tal i com s'ha explicat en l'apartat del timbre, el so d'una nota està compostat, per moltes més freqüències que la fonamental, amb la qual cosa faltaria fer un estudi harmònic de les diferents notes, tal i com es fa en l'**Annex E**, i efectivament, observar que hi ha una important variació entre les composicions harmòniques.

Tal i com s'ha vist a la *figura 3* i com s'ha explicat a l'apartat del timbre, és impossible esperar la mateixa entrada de freqüències per a una mateixa nota, inclús s'ha analitzat en estudis que fent proves amb el mateix instrument el timbre pot variar en funció de les diferències antropomètriques de l'interpret[9].

Per simplificar aquest problema es decideix crear aquest a suma freqüencial del clarinet de forma sintètica, amb una mateixa funció que crea la nota amb els 7 primers harmònics. Un cop tenim la funció de síntesis es tracta de implementar-la per a cada una de les freqüències fonamentals de la tessitura del clarinet, i d'aquesta manera sumar tots els harmònics resultants obtinguts.

Aquest és un model que inicialment es va dissenyar pensant que serviria com a filtre, o funció de transferència que representa el clarinet, i que a partir de qualsevol nota entrada d'una trompeta, per exemple, en sortiria la seva corresponent amb timbre de clarinet.

Efectivament, és un disseny invàlid, per una raó tan bàsica com que qualsevol dels harmònics que conté una nota també pot formar part d'una altra però en diferent amplitud i aquest filtre que s'ha dissenyat no fa distinció de la nota d'entrada, i acaba sent un filtre que permet la sortida d'un cúmul d'harmònics sense complir cap relació entre ells i amb amplituds resultants de la suma de varies amplituds diferents dels harmònics amb els que s'ha creat.

7.3 Segon Disseny: Utilitzant *Cross-Synthesis*

El *cross-Synthesis* és un mètode de síntesis que es basa en l'encreuament de diferents sons. Dit d'una altra manera, permet aplicar paràmetres com l'amplitud o freqüències d'un so o senyal moduladora sobre un altre o senyal portadora.

El procediment tècnic del *cross-Synthesis* entre dos senyals es pot resumir amb:

- Realitzar la Transformada de Fourier en Temps Discret, de les dos senyals, la moduladora i la portadora.
- Calcular l'envolvent espectral en el marc temporal per cada una.
- Dividir l'espectre de la portadora per cada mostra del seu propi envolvent per aplanar-la.
- Multiplicar mostra a mostra d'aquest nou envolvent aplanat pel de la moduladora, d'aquesta manera es reemplaça l'envolvent de la portadora pel de la moduladora.

Un clar exemple en son els *vocoders* comercials, com el de la *figura 42*, que consten d'un teclat sintetitzador, que s'encarrega de generar el timbre desitjat, i un micròfon acoblat, que permet captar veu i transformar el so anteriorment esmentat. D'aquesta manera s'aconsegueix un efecte similar a un "orgue que parla".

Un altre exemple on s'utilitza *cross-Synthesis* és en un procés de *morphing*, que equival a canviar gradualment d'un so a un altre mitjançant atenuació i mescla dels dos senyals.

Tot i que generalment per fer aquestes transicions sonores més acurades s'utilitzen tècniques d'interpolació, o processos estocàstics[10], per veure un exemple senzill i poder provar-ho s'ha realitzat un típic *crossfade* entre dos senyals tal i com mostra el següent codi.

```
fFade = font(fSamples+1:length(font)); % s'extreu el bloc de senyal source que interessa
oFade = objectiu(1:length(objectiu)-oSamples); % s'extreu el bloc de senyal target que interessa

fEnv = linspace(1, 0, length(fFade)); % envolvent d'esvaiment per al senyal source
oEnv = linspace(0, 1, length(oFade)); % envolvent de creixement per al senyal target

% s'apliquen els envolvents als respectius senyals i es combinen
fCombine = [fEnv.*fFade zeros(1,dSamples-length(fFade))];
oCombine = [zeros(1, dSamples-length(oFade)) oEnv.*oFade];
dCombined = fCombine + oCombine;

y = [font(1:fSamples) dCombined objectiu(length(oFade)+1:length(objectiu))]; % senyal final
```

On: **font** es el primer senyal, **objectiu** es el segon senyal, **fSamples** son les mostres amb només senyal font, **oSamples** son les mostres amb només senyal objectiu i **dSamples** les mostres on s'entrecreuen els 2 senyals.

La idea és la d'aprofitar aquest encreuament dels dos senyals on es combinen i creen un nou so. Però això, aplicat a l'objectiu de transformar un timbre, a partir d'una font inicial, en un altre d'específic, no resulta gens satisfactori ja que la seva mescla d'harmònics impossibilita la definició correcta del timbre objectiu.

Arribat en aquest punt s'han de buscar mètodes on es faci una distinció de la nota interpretada, per poder llavors aplicar un processat concret i específic.

7.4 Tercer Disseny: Utilitzant MIDI

MIDI és l'acrònim de Musical Instrument Digital Interface, en català, interfície digital d'instruments musicals. És un protocol de comunicació estàndard que permet a les computadores, sintetitzadors, seqüenciadors, controladors i altres dispositius o instruments musicals electrònics comunicar-se i compartir informació per el treball amb la música i el so.

Aquesta informació que intercanvien aquests aparells acostuma a contenir notes, accions o esdeveniments, i valors per a controladors, i es caracteritza per la seva senzillesa, el poc ús de dades i espai necessari, la seva escalabilitat i compatibilitat en tot tipus de instruments i també pot servir com a ús de composició i d'escriptura de partitures digitals.

El MIDI té un paper molt important en el món dels sintetitzadors, i de la síntesis processada en computadores. Mitjançant un teclat o piano digital que utilitzi MIDI connectat a una computadora, podem utilitzar-lo com a controlador, es a dir, l'aparell deixa de generar so per si mateix, d'aquesta manera la computadora és qui s'encarrega de reproduir el so sintetitzat, i d'aquesta manera abaratir el cost de molts instruments musicals, o permetre una ampliació de les seves capacitats amb la compaginació d'una computadora. Un altre exemple podria ser un *sampler*, o aparell que es dedica a "disparar" sons prèviament emmagatzemats. Mitjançant midi podem convertir qualsevol dispositiu capaç de comunicar-se amb una computadora per tal de "disparar" aquests sons. El mateix passa amb sintetitzadors, el MIDI permet controlar un sintetitzador software que es qui processa la síntesis a temps real.

En aquesta opció de disseny, el MIDI entra al procés com a controlador o disparador d'esdeveniments. Si l'objectiu és el de transformar timbre, i s'ha estudiat la variació de forma espectral en relació a les diferents notes, es podria considerar algun tipus de mecanisme i muntatge electrònic, tipus sensor, que ensamblat al mateix instrument amb el que s'emet el so original, permeti enviar la informació de la nota que s'està tocant, i així doncs fer un tractament de la senyal adequat al so per cada un dels inputs MIDI. El benefici d'aquest procediment és l'estalvi de l'anàlisi previ del so abans de fer les transformacions adients, i així doncs rapidesa en el procés general gracies a l'automatització.

7.5 Quart Disseny: Utilitzant el Model d'Anàlisi/Síntesi Sinusoïdal + Soroll

7.5.1 Introducció a la Detecció de Pitch

Les tècniques de detecció de pitch, o altrament dit, d'estimació de freqüència fonamental (F_0), són tècniques que permeten extreure el to d'un senyal gairebé periòdic, normalment d'enregistraments digitals d'interpretacions musicals, o veu.

El perfeccionament d'aquestes tècniques permet als algorismes dur a terme la detecció en temps real. El que distingeix si l'algorisme és a temps real és el fet de que tardi menys en calcular el to d'un segment de so que no pas el temps que duri aquest segment.

Existeixen diversos sistemes interessants de detecció de *pitch*[11]. Alguns dels mes utilitzats son:

- Detecció de creuament per zero.
- Detecció de pics.
- Comparació de les formes d'ona.
- Autocorrelació
- Espectrografia.
- Cepstrum.

Tot i així com que aquesta extracció de F_0 pot ser en àmbits i situacions molts diferents, es troben altres variants, amb millores o mètodes combinats.

Pel cas d'una extracció de F_0 en una suposada melodia de clarinet, els mètodes que s'han determinat com a més interessants son el d'autocorrelació, el que utilitza la transformada discreta de Fourier STFT i el mètode HPS (Harmonic Product Spectrum).

El mètode d'autocorrelació, que actua basant-se en anàlisis al domini temporal, dona molt bons resultats a freqüències baixes, ja que la seva resolució depèn de la freqüència de mostreig, i com més baixes freqüències, més llarga la longitud d'ona amb la qual cosa deixa un marge d'error més petit.

Per altra banda utilitzar la STFT és interessant per a freqüències altes. Aquest mètode actua en el domini freqüencial, i la seva resolució es constant en relació a les freqüències, però com que a les notes musicals més agudes la diferencia de potencia entre freqüència fonamental i harmònics és més gran, hi ha menys risc d'error.

I l'últim, el mètode HPS aconsegueix una bona distinció entre freqüència fonamental i els harmònics que pugui tenir el timbre del propi instrument[12].

Així doncs es proposa un algorisme híbrid que determini el to mitjançant aquests tres mètodes anteriorment esmentats tot aconseguint una millora substancial pel nostre cas particular:

- Es realitza FFT
- Amb el mètode HPS es calcula la fonamental, però només serveix per una estimació inicial. La resolució és baixa perquè depèn de la freqüència de mostreig i del nombre de mostres de l'espectre, que no pot ser molt gran per no elevar el cost de processament i causar retard a l'operació.
- Si la F_0 obtinguda anteriorment és baixa, s'utilitza autocorrelació per obtenir una resolució millor.
- Si es alta, es realitza la STFT.

Un cop en execució, paral·lelament es manté el càlcul mitjançant HPS, i cada cert temps, gràcies a l'estimació d'aquest, es verifica si es continua amb l'actual, o es canvia a l'oposat d'entre els mètodes destinats a freqüències agudes o greus.

7.5.2 Model d'Anàlisi / Síntesi Sinusoidal + Soroll

Aquest cinquè disseny utilitza un sistema d'anàlisi i síntesi amb molt de potencial que es basa en la descomposició del senyal en part harmònica i part residual, exposat i exemplificat a l'**Annex F**.

Per entendre el seu funcionament en tenim un exemple utilitzat per a anàlisi i síntesi de veu a la *figura 50*. En aquesta s'observa com a partir d'una entrada de veu real, se n'obté una de sintetitzada, fent doncs una re-síntesi additiva.

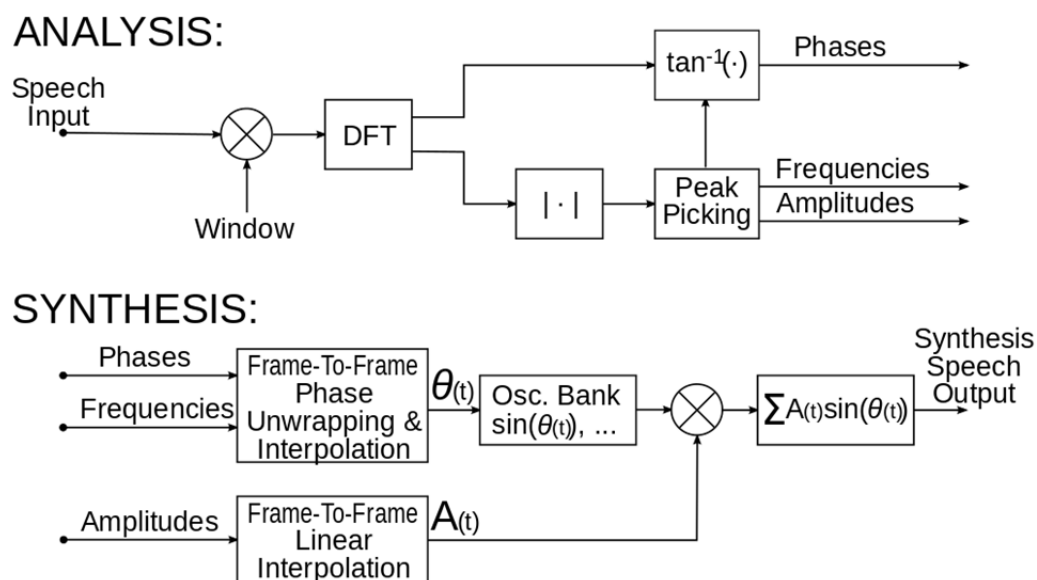


Fig. 50 Sistema d'Anàlisi/Síntesi per al model Sinusoidal basat en McAulay & Quatieri.

La part més interessant d'aquest mètode per aquest cinquè disseny és el punt en el que tenim una descomposició sinusoidal de la part harmònica. Això és precisament el que ens interessa tractar i modificar-ho per tal d'alterar el timbre de l'instrument. A més a més es conserva la part no harmònica, o soroll, per si a l'hora de construir e senyal final es vol afegir per mantenir matisos no musicals, però que li donen un aspecte de so mes realista.

El funcionament d'aquest disseny aprofitaria totes i cada una de les parts del model d'anàlisi/síntesi sinusoidal + soroll, però cal afegir els processos adients per complir amb els objectius.

Juntament amb l'anàlisi inicial afegirem l'algorisme proposat en l'apartat anterior per a la detecció de la freqüència fonamental i un cop obtinguda servirà per aplicar-hi un filtre pertinent. Es fa la descomposició de part sinusoidal + soroll, es fa la re-síntesi de la part harmònica i se li aplica el filtre anteriorment anomenat, d'aquesta manera filtrem harmònics purs i imprescindibles cada un d'ell per al timbre desitjat. Aquests filtres estarien parametritzats en el que podria ser un banc de filtres, on cada freqüència fonamental d'entrada possible, és a dir cada nota de l'instrument, tindria associat un filtre dissenyat específicament que tindrà en compte les estructures harmòniques origen i destí. D'aquesta manera la variació d'amplituds entre notes no suposa cap problema. Per últim es pot afegir el soroll extret (en el qual també s'hi poden aplicar modificacions) fent una segona re-síntesi. Es mostra l'esquema a la *figura 51*.

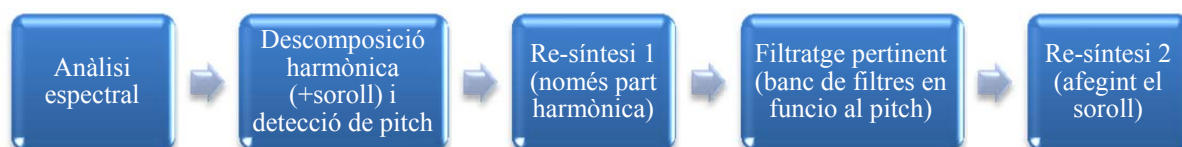


Fig. 51 Esquema disseny utilitzant model de descomposició harmònica + soroll.

7.6 Cinquè Disseny: Utilitzant Síntesi per Modelatge Físic

El cinquè disseny es basaria en la síntesi per modelatge físic, i mitjançant una taula d'equivalències de paràmetres entre el timbre d'entrada i el de sortida.

Per fer això es obligatori disposar dels dos models, mitjançant un anàlisi previ determinar la freqüència fonamental, per després fer el procés invers en el model de l'instrument font, en aquest cas del clarinet, per obtenir els paràmetres i buscar el seu equivalent per al model del timbre destí tal i com mostra el diagrama de blocs de la *figura 52*.

Aquest procés invers ens donaria paràmetres com ara la pressió teòrica del flux d'aire que estem aplicant, i gràcies a la taula de conversió es podria complementar aquesta dada aplicada al model

del instrument resultant, que tot i ser diferent per una conversió més correcta és necessari que sigui de la mateixa família, en aquest cas de vent per poder mantenir una correlació entre paràmetres.

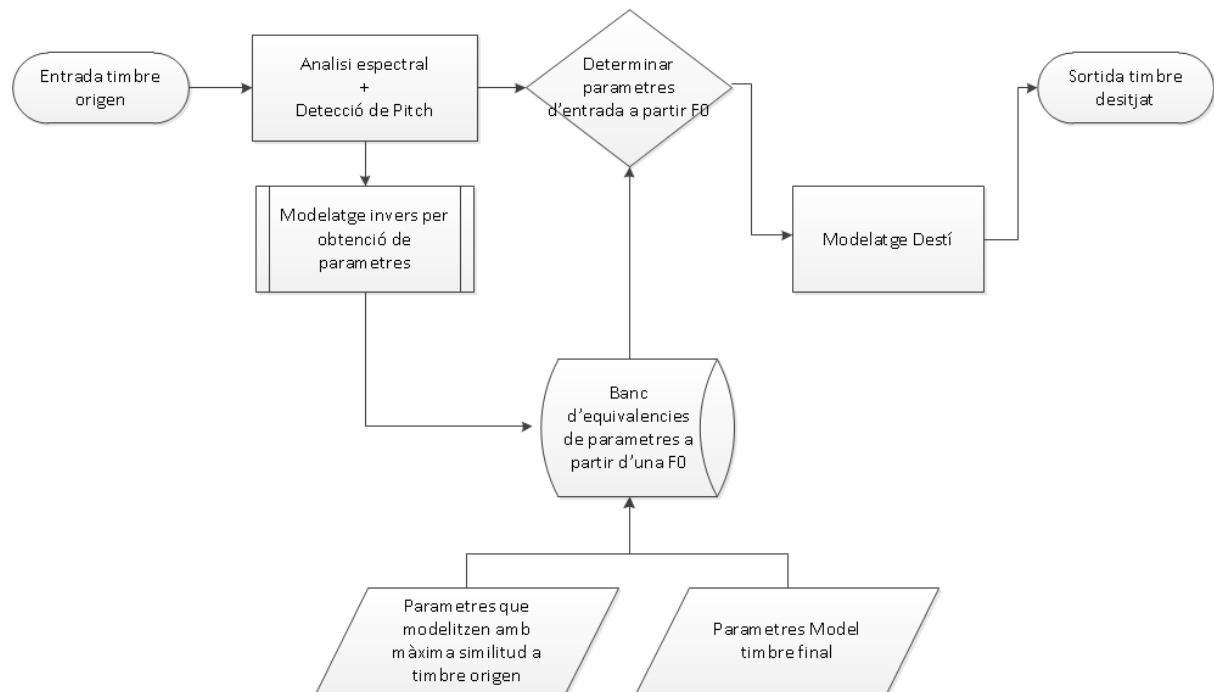


Fig. 52 Diagrama de blocs per a un disseny de transformació a partir de síntesis per modelatge físic.

8 Conclusions

8.1 Conclusions

Tal i com es detalla a l'inici, a l'apartat d'objectius plantejats, en aquest projecte es volia dissenyar un sistema que permetés transformar timbre en un instrument determinat, analitzant-lo prèviament.

Un cop finalitzat el projecte, es disposa d'un gran estudi de cada una de les parts teòriques que fonamenten el tractament de so concretant cada vegada més fins arribar a la modificació tímbrica. Això ha suposat un procés d'investigació molt important, allargant doncs, molt més la part teòrica i de coneixements necessaris. Però es disposa també de diferents dissenys propis basats en l'aprofitament de metodologies diverses i que son amb els quals ha evolucionat aquest estudi, i s'ha perfeccionat respecte als plantejaments inicials.

MatLab ha estat des de l'inici el software escollit per dur a terme anàlisis, filtratges, probes i petites implementacions que han servit per validar o reconduir el camí a seguir. Tot i així, ha estat a la part final de la realització d'aquest projecte quan s'han descobert noves eines més específiques i còmodes, ja que MatLab està bastant limitat per a procediments a temps-real, amb les quals poder implementar un sistema com el desitjat. Al següent apartat de treball futur es detallen algunes d'aquestes eines.

Pel que fa al gran bloc de tipus de síntesis, veure totes aquestes maneres diferents de recrear sons desitjats, ha estat la peça clau per entendre bé el concepte de síntesi, arribant a la conclusió de que tot i necessitar més recursos de processament, la síntesis additiva és la base, la més trivial, amb la que s'obté més precisió i per això la més interessant en una transformació tímbrica concreta. També s'ha entès la importància i la relació entre un anàlisi, previ, i una síntesi posterior, per a les transformacions sonores. És a dir, el perquè cal sintetitzar si l'únic objectiu es modificar el so, i la resposta més clara la trobem en el quart disseny, el que es basa en el model d'anàlisi/síntesi sinusoïdal + soroll.

El quart disseny, és amb el que concloc, ja que és per mi el disseny més òptim i que permet un major control sobre la manipulació tímbrica. L'Annex F exemplifica perfectament el funcionament del model d'anàlisi/síntesi sinusoïdal + soroll, i és quan es veu la possibilitat de re-sintetitzar de la forma desitjada mitjançant el control harmònic, on entra en joc la modificació tímbrica, tal i com s'ha descrit a l'apartat d'aquest quart disseny.

Per tant doncs es pot dir que els objectius han estat assolits, tot i així queden diversos camins per on continuar i seguidament és detalla el treball que precedirà.

8.2 Treball Futur

La principal tasca que es planteja per abordar en un futur proper és la d'implementar mitjançant una aplicació el disseny escollit com a més apte. A part del software analitzat anteriorment, i un cop conegut el disseny dels diferents algorismes que compleixen amb l'objectiu de fer aquestes transformacions tímbriques, s'han descobert altres eines que treballen a un nivell de programació més alt, però que per altra banda són més específiques i presumeixen d'una bona documentació. Dues d'aquestes possibles eines podrien ser:

- SMS-tools del Music Technology Group de la Universitat Pompeu Fabra. Impulsat per Xavier Serra[13], ofereix una sèrie de tècniques i implementacions adients per a l'anàlisi, la transformació i la síntesi de sons musicals en base a diversos mètodes de modelització espectral. Aporta també l'opció de treball en temps real mitjançant programació amb *Python*.
- *Loris*, un paquet de software lliure exprés per al modelatge, la transformació i la manipulació del so en temps real. Aquesta interfície i les seves llibreries suporten els llenguatges *C*, *C++* i *Python* i aporten una bona documentació per cada cas.

Un altre camp per on explotar aquests coneixements, és a través del mòbil, que no deixa de ser una eina de processament portàtil i cada cop més potent. Els micròfons que porten integrats no son gens adients per una adquisició de senyal auditiu, però amb ajuda de micròfons externs connectats al dispositiu pot resultar molt interessant la implementació d'aplicacions d'aquest àmbit de processament del senyal. Una eina perfecta per aportar tot el potencial de *PureData* al mòbil, es la llibreria [*Libpd*](#), que consta d'implementacions de *PD* per a *Android* i *iOS* entre d'altres llibreries.

9 Bibliografía

- [1] T. J. Ulrich, “Envelope calculation from the Hilbert transform,” 2006.
- [2] J. O. S. Iii, “Viewpoints on the History of Digital Synthesis * Historical View of Synthesizer Development,” in *International Computer Music Conference*, 2005, pp. 1–10.
- [3] G. Marentakis and K. Jensen, “Sinusoidal Synthesis Optimization,” vol. 4, no. 1, pp. 1–4, 2000.
- [4] J. M. Chowning, “The synthesis of complex audio spectra by means of frequency modulation,” *J. audio Eng. Soc.*, 1973.
- [5] P. Guillemain, J. Kergomard, and T. Voinier, “Real-time synthesis of clarinet-like instruments using digital impedance models,” *J. Acoust. Soc. Am.*, vol. 118, no. 1, p. 483, 2005.
- [6] Julius O. Smith III, *Physical Audio Signal Processing*. W3K Publishing, 2010.
- [7] J. Smith, “Physical modeling using digital waveguides,” *Comput. Music J.*, 1992.
- [8] D. . R. M . Rath , F . Avanzini , N . Bernardini , G . Borin , F . Fontana , L . Ottaviani, “An introductory catalog of computer-synthesized contact sounds, in real-time,” in *Colloquium, X I V Informatics, Musical Cim, X I V*, 2003, pp. 1–6.
- [9] F. G. M. A, L. De Acústica, S. De Música, and E. Martínez, “Variaciones en el timbre del clarinete causadas por las diferencias antropométricas de dos intérpretes,” in *TecniAcustica*, 2003.
- [10] J. Maccallum, A. Schmeder, and D. Wessel, “Timbral migration: stochastic processes for the control of smooth spectral transformation,” 2003.
- [11] D. Gerhard, *Pitch extraction and fundamental frequency: History and current techniques*. 2003, pp. 0–22.
- [12] P. D. La Cuadra, “Efficient pitch detection techniques for interactive music,” ... *Comput. Music ...*, 2001.
- [13] X. Serra, “A system for sound analysis / transformation / synthesis based on a deterministic plus stochastic decomposition,” Stanford, 1989.

Annex A – Relació Entre Freqüència i Notes Musicals

Nota (notació anglosaxona)	Freqüència (Hz)	Longitud d'ona (cm)
C ₀	16.35	2100
C [#] ₀ /D ^b ₀	17.32	1990
D ₀	18.35	1870
D [#] ₀ /E ^b ₀	19.45	1770
E ₀	20.6	1670
F ₀	21.83	1580
F [#] ₀ /G ^b ₀	23.12	1490
G ₀	24.5	1400
G [#] ₀ /A ^b ₀	25.96	1320
A ₀	27.5	1250
A [#] ₀ /B ^b ₀	29.14	1180
B ₀	30.87	1110
C ₁	32.7	1050
C [#] ₁ /D ^b ₁	34.65	996
D ₁	36.71	940
D [#] ₁ /E ^b ₁	38.89	887
E ₁	41.2	837
F ₁	43.65	790
F [#] ₁ /G ^b ₁	46.25	746
G ₁	49	704
G [#] ₁ /A ^b ₁	51.91	665
A ₁	55	627
A [#] ₁ /B ^b ₁	58.27	592
B ₁	61.74	559
C ₂	65.41	527
C [#] ₂ /D ^b ₂	69.3	498
D ₂	73.42	470
D [#] ₂ /E ^b ₂	77.78	444
E ₂	82.41	419
F ₂	87.31	395
F [#] ₂ /G ^b ₂	92.5	373
G ₂	98	352
G [#] ₂ /A ^b ₂	103.83	332
A ₂	110	314
A [#] ₂ /B ^b ₂	116.54	296
B ₂	123.47	279
C ₃	130.81	264
C [#] ₃ /D ^b ₃	138.59	249
D ₃	146.83	235
D [#] ₃ /E ^b ₃	155.56	222
E ₃	164.81	209

Nota (notació anglosaxona)	Freqüència (Hz)	Longitud d'ona (cm)
F ₃	174.61	198
F [#] ₃ /G ^b ₃	185	186
G ₃	196	176
G [#] ₃ /A ^b ₃	207.65	166
A ₃	220	157
A [#] ₃ /B ^b ₃	233.08	148
B ₃	246.94	140
C ₄	261.63	132
C [#] ₄ /D ^b ₄	277.18	124
D ₄	293.66	117
D [#] ₄ /E ^b ₄	311.13	111
E ₄	329.63	105
F ₄	349.23	98.8
F [#] ₄ /G ^b ₄	369.99	93.2
G ₄	392	88
G [#] ₄ /A ^b ₄	415.3	83.1
A ₄	440	78.4
A [#] ₄ /B ^b ₄	466.16	74
B ₄	493.88	69.9
C ₅	523.25	65.9
C [#] ₅ /D ^b ₅	554.37	62.2
D ₅	587.33	58.7
D [#] ₅ /E ^b ₅	622.25	55.4
E ₅	659.26	52.3
F ₅	698.46	49.4
F [#] ₅ /G ^b ₅	739.99	46.6
G ₅	783.99	44
G [#] ₅ /A ^b ₅	830.61	41.5
A ₅	880	39.2
A [#] ₅ /B ^b ₅	932.33	37
B ₅	987.77	34.9
C ₆	1046.5	33
C [#] ₆ /D ^b ₆	1108.73	31.1
D ₆	1174.66	29.4
D [#] ₆ /E ^b ₆	1244.51	27.7
E ₆	1318.51	26.2
F ₆	1396.91	24.7
F [#] ₆ /G ^b ₆	1479.98	23.3
G ₆	1567.98	22
G [#] ₆ /A ^b ₆	1661.22	20.8
A ₆	1760	19.6

Annex B – Equalitzador de 3 Bandes

```
%%Equalitzador de 3 bandes
[x, fs]=wavread('Fixer_Orig.wav');
N = 100; % Ordre dels filtres FIR

% Els guanys per cada una de les 3 bandes
gPB = 1; %0.8;
gPM = 5; %1.5;
gPA = 0.2; %1.3;

Wn = .07; % Freqüència de tall normalitzada
PB = fir1(N,Wn); % Filtre Passa-Baixos
Wn1 = [.07, .10];
PM = fir1(N,Wn1); % Filtre Passa-Banda
Wn2 = .10;
PA = fir1(N,Wn2,'high'); % Filtre Passa-Alts

figure(1);
freqz(PB);
figure(2);
freqz(PM);
figure(3);
freqz(PA);

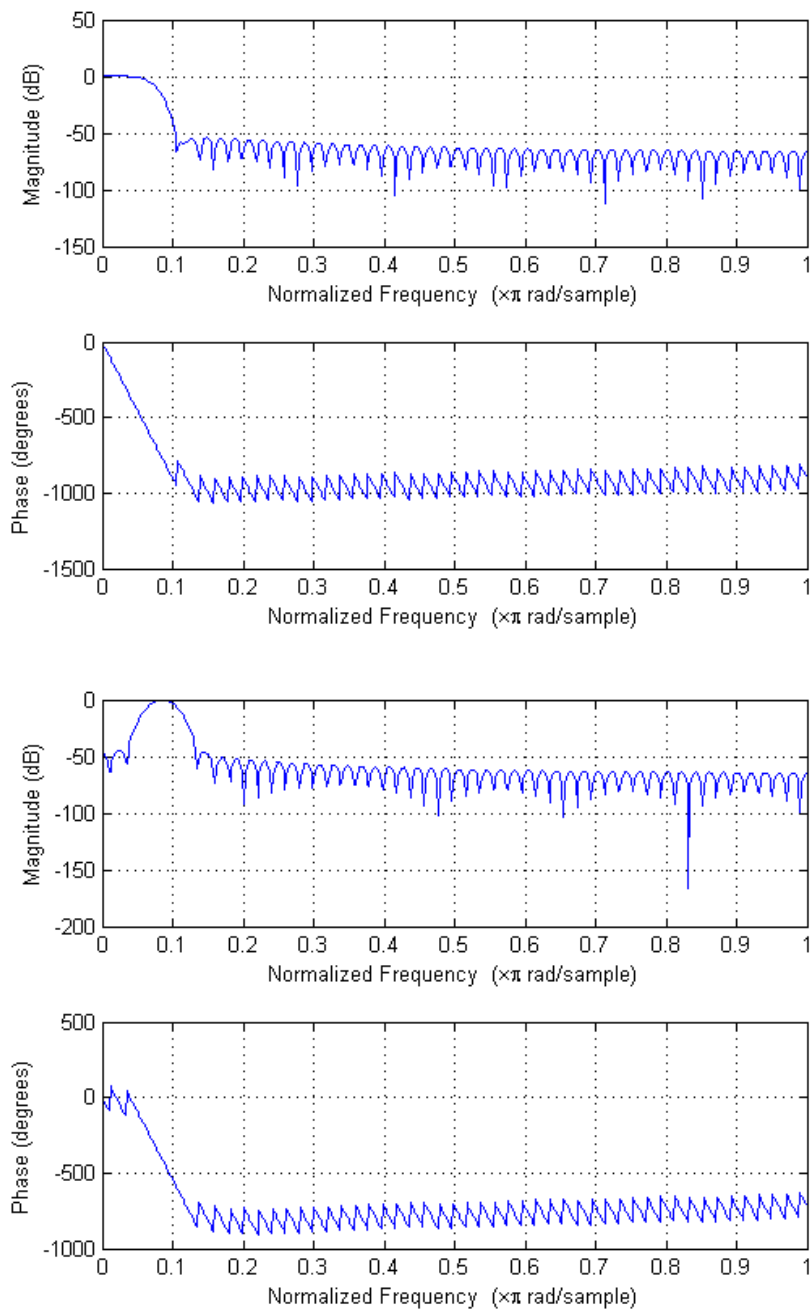
y1 = conv(PB,x);
y2 = conv(PM,x);
y3 = conv(PA,x);

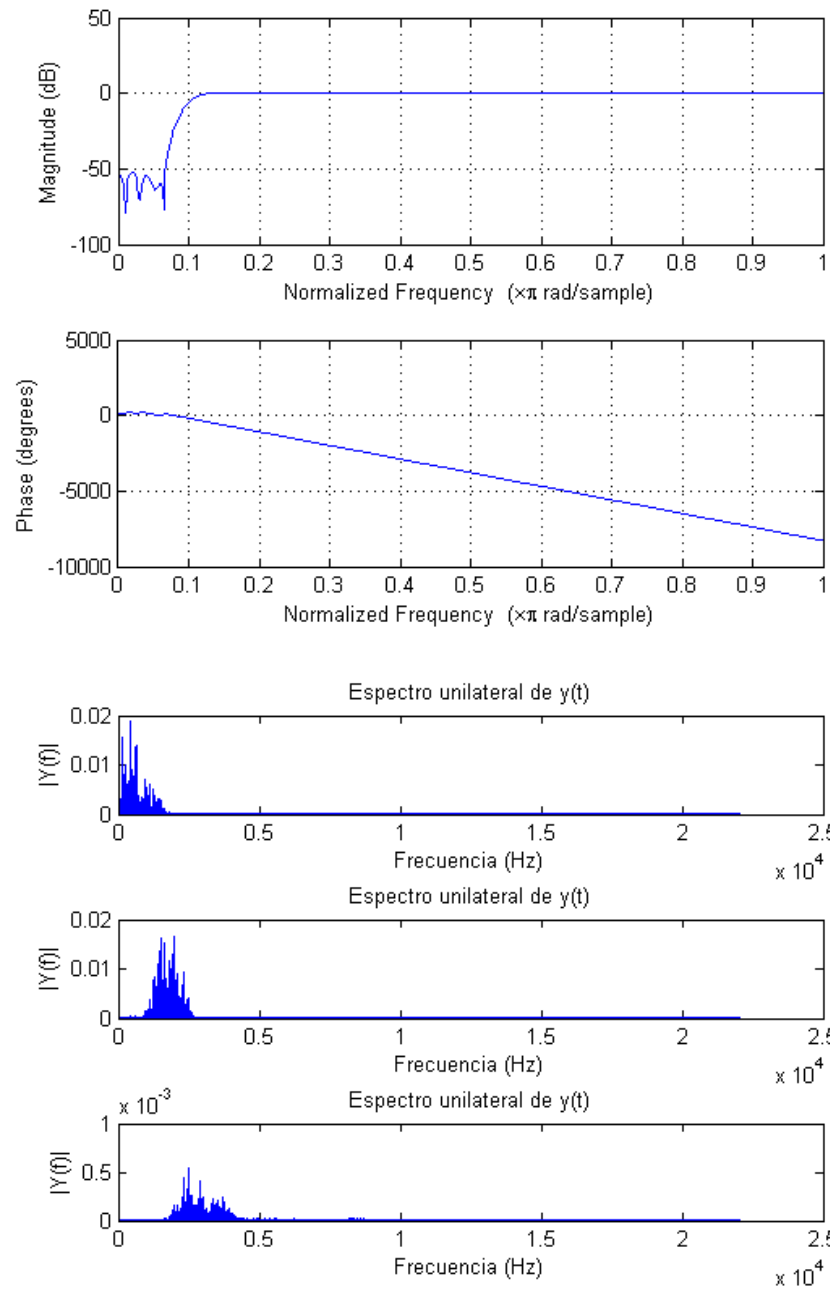
yA = gPB * y1;
yB = gPM * y2;
yC = gPA * y3;
yD = yA + yB + yC;

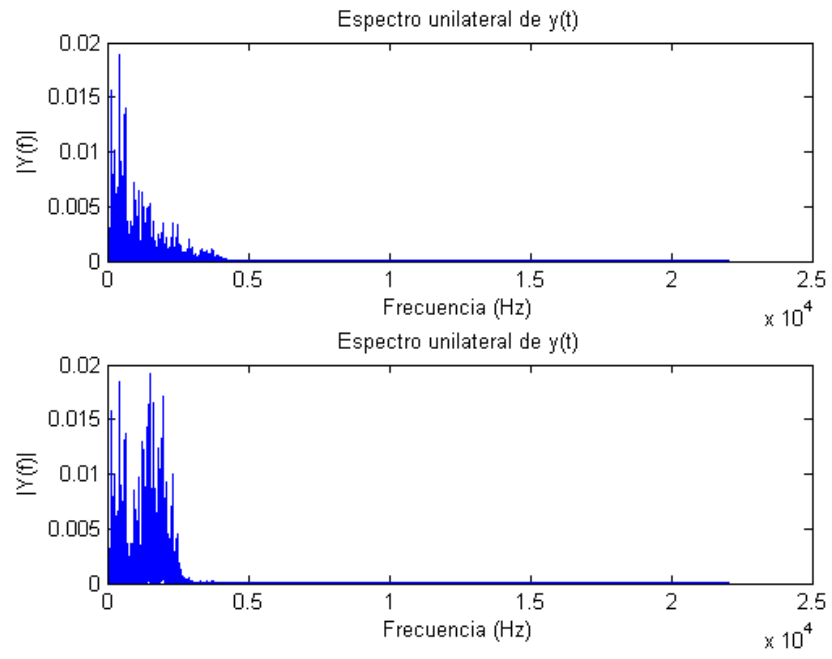
% Gràfiques dels 3 espectres amb filtre i guany aplicats
figure(4);
subplot(3,1,1);
fft_signal(yA,fs);
subplot(3,1,2);
fft_signal(yB,fs);
subplot(3,1,3);
fft_signal(yC,fs);

% Gràfica comparació espectre senyal original amb la equalitzada
figure(5);
subplot(2,1,1);
fft_signal(x,fs);
subplot(2,1,2);
fft_signal(yD,fs);

% S'escriuen els fitxers d'audio parcials i el final
wavwrite(yA,fs,'Equalitzat_PB');
wavwrite(yB,fs,'Equalitzat_PM');
wavwrite(yC,fs,'Equalitzat_PA');
wavwrite(yD,fs,'Equalitzat_TOTAL');
```







Annex C – Funció d’Envolvent a partir de la Transformada de Hilbert

```
function [env, ht] = hilbert_envelope(wf, varargin)
% Author: TJ Ulrich 5/23/05, updated 3/20/06 with code from RAG, see **
% below.
% Purpose: create the envelope function of a given waveform, also provide
% the hilbert transform of the original waveform
% Parameters: Input: wf = waveform to be enveloped (preferably has a
% length that is a power of 2)
% sp = flag to indicate which method to use, i.e. is
% the signal processing toolbox present.
% 0 = no (DO NOT use the signal proc. toolbox)
% 1 = yes (use Sig. Proc. toolbox)
% Output: env = envelope to be returned
% ht = Hilbert transform of wf
%
% ** the code from Robert Guyer was used/altered to make this function not
% dependent upon the availability of the signal processing toolbox. The
% functionality using the sig. proc. toolbox is still available using a 1
% as the second input parameter. a 0 or no second parameter will neither
% require nor use the signal processing toolbox.
%
% Syntax: env = envelope(wf) will produce the envelope of the waveform wf
% without using the signal processing toolbox.
% [env, HT] = envelope(wf) same as above but will also output the
% hilbert transform in the variable HT.
% env = hilbert(wf,1) same as env = hilbert(wf) but WILL use the
% signal processing toolbox (i.e., the hilbert() function)
% [env, HT] = envelope(wf,1) same as above but will also output the
% hilbert transform in the variable HT.

% check for additional arguments, i.e. whether or not to use signal
% processing toolbox functions
if nargin > 1
    sp = varargin{1};
else
    sp = 0; % default value
end
% check for valid values of sp
switch sp
case 0 % do nothing (valid value)
case 1 % do nothing (valid value)
otherwise
    sp = 0; % change to default valid value
end

% ----- create the envelope and Hilbert transform -----
if sp % use the signal processing toolbox
    env = abs(hilbert(wf));
    ht = imag(hilbert(wf));
else
    %%%%%%%%%
    Nfft=length(wf);
    sz = size(wf); % get size of wf ... i.e. row or column vector
    h1fsz1 = ceil(sz/2); % get half the size (use ceil() for odd # of points)
    h1fsz2 = floor(sz/2); % get half the size (use floor() for odd # of points)
    h1fsz1(h1fsz1<1) = 1; % fix fractional dimensions
    h1fsz2(h1fsz2<1) = 1; % fix fractional dimensions
    %%%%%%%%% FFT
```



```

FFTf=fft(wf,Nfft);
%%%%%% Hilbert transform
npl us=(1:ceil(Nfft/2));
nmi nus=(1+ceil(Nfft/2):Nfft);
Hfactor=zeros(sz);
Hfactor(npl us)=ones(hl fsz1);
Hfactor(nmi nus)=-ones(hl fsz2);
%%%%%%
Hf=Hfactor.*FFTf;
%%%%%% in time domain
f1=i fft(FFTf,Nfft);
h1=i fft(Hf,Nfft);
ht = imag(h1);
%%%%%% calculate envelope
e1=f1+h1;
env=sqrt(e1.*conj(e1));
end

```

Annex D – Comparativa Síntesi Entre MAX, PD, SuperCollider i Reaktor

Per poder tenir una primera impressió d'aquests programes i veure les diferències i similituds entre el seus sistemes de programació, es mostra una síntesi additiva d'ona quadrada simplificada amb els sis primers harmònics, i implementada de la mateixa manera en cada un d'ells.

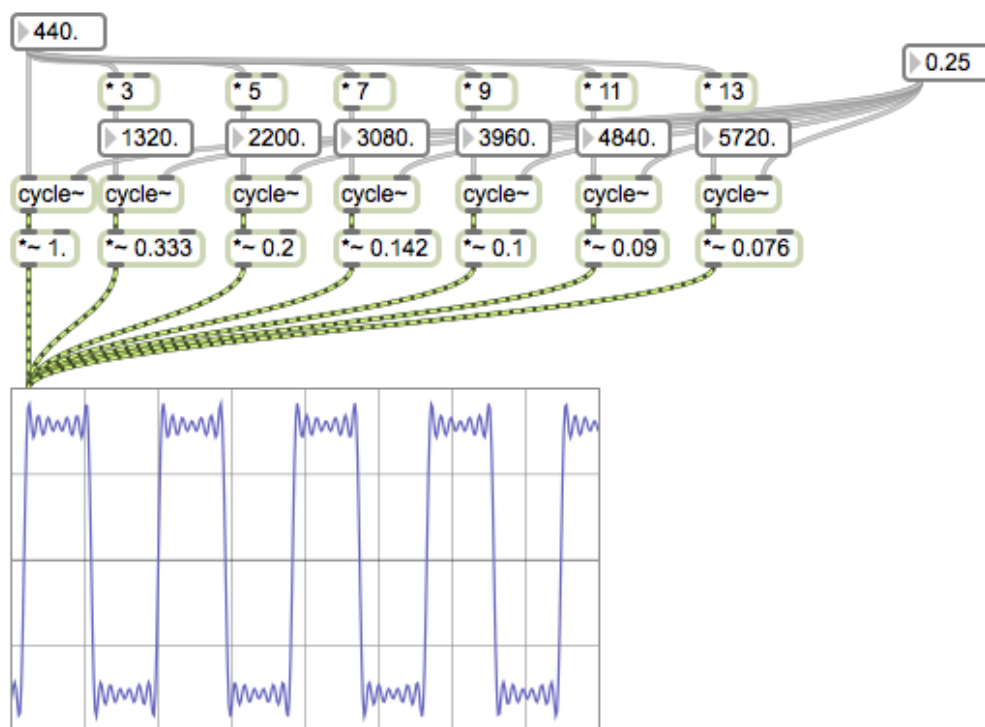
La composició d'aquesta ona quadrada és a base d'harmònics imparells amb una amplitud respectiva que decreix inversament proporcional a la freqüència dels harmònics, es a dir:

$$\text{Ona quadrada} = (F_0, A_0) + (3F_0, \frac{A_0}{3}) + (5F_0, \frac{A_0}{5}) + (7F_0, \frac{A_0}{7}) + \dots$$

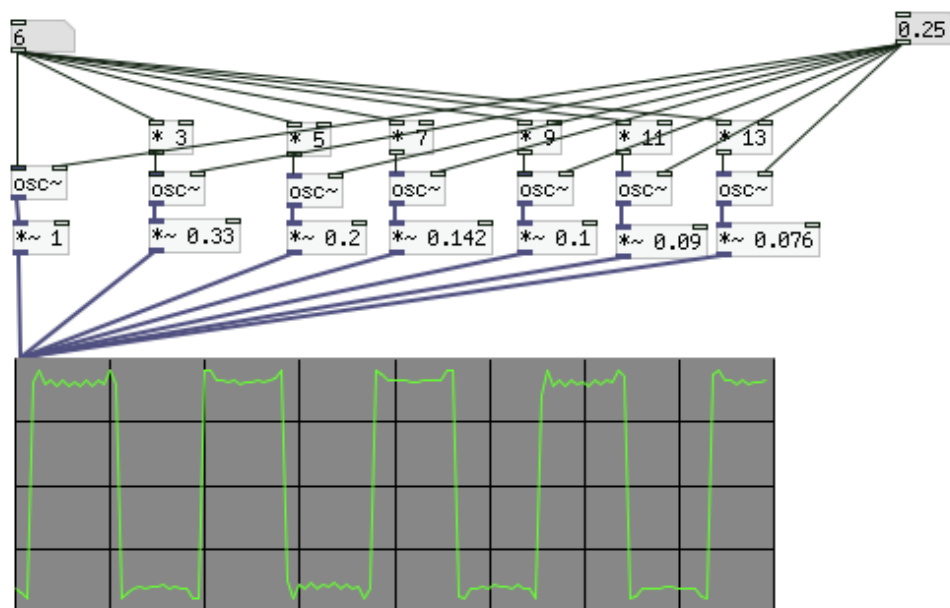
On F_0 és la freqüència fonamental i A_0 l'amplitud normalitzada.

Com que amb Max i amb PD els objectes oscil·ladors son amb la funció cosinus, s'afegeix un desplaçament de 0,25 cicles per que la seva visualització concordi amb els altres exemples que utilitzen la funció amb el sinus.

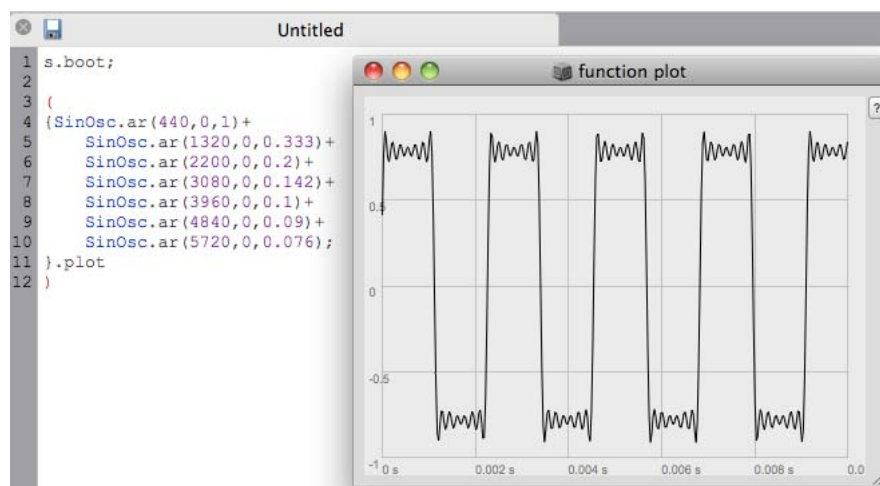
Exemple amb Max:



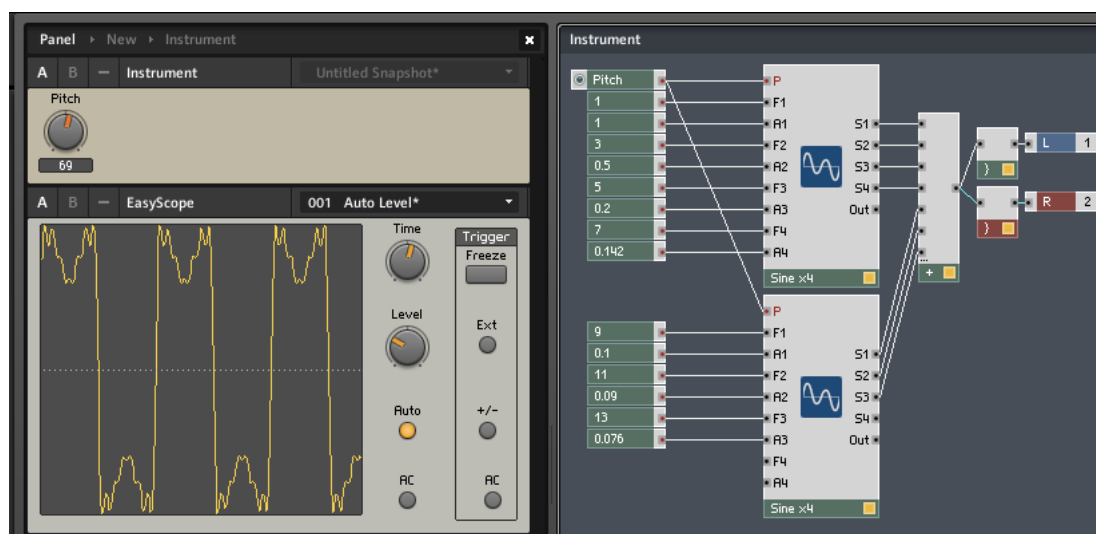
Exemple amb PD:



Exemple amb SuperCollider:



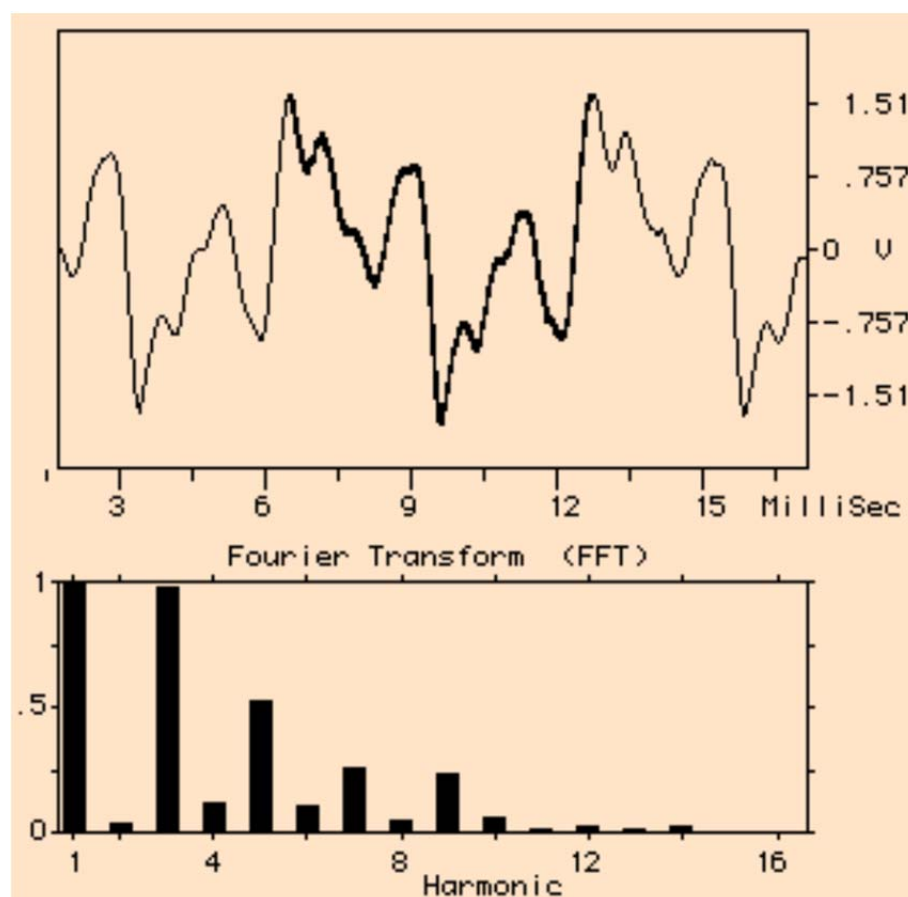
Exemple amb Reaktor:



Annex E – Comparativa Freqüencial en Notes del Clarinet

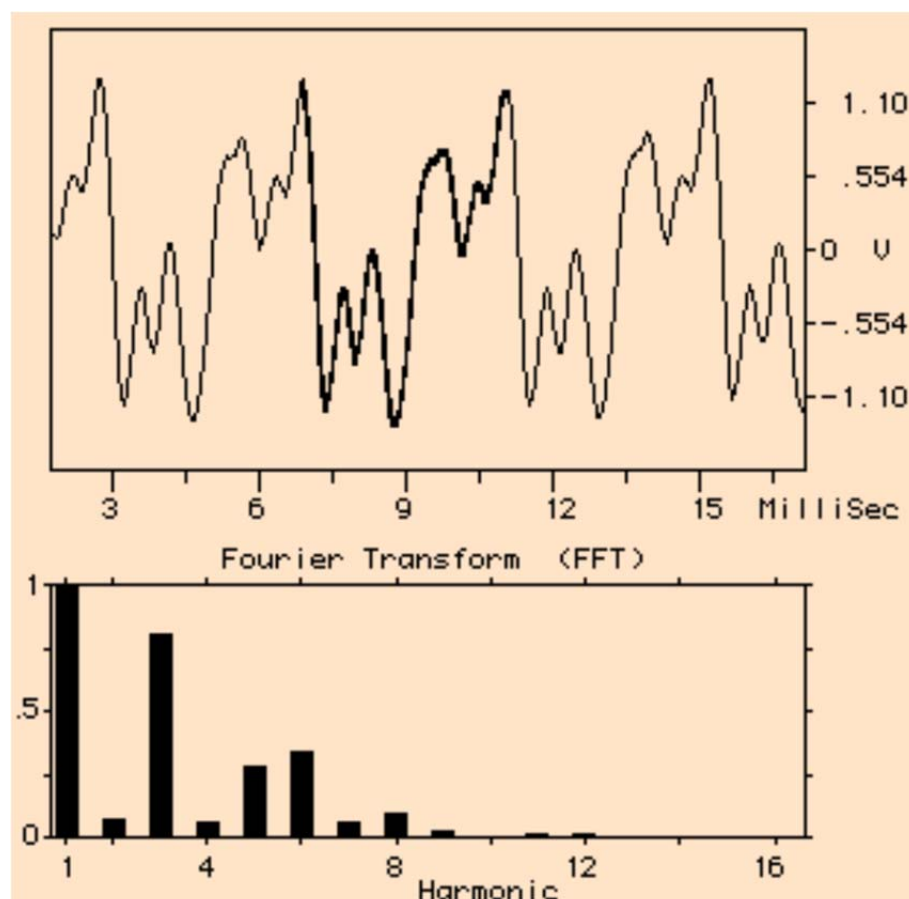
- Nota Eb3 a 156 Hz

Harmònic	Amplitud
1	1.00
2	0.04
3	0.99
4	0.12
5	0.53
6	0.11
7	0.26
8	0.05
9	0.24
10	0.07
11	0.02
12	0.03
13	0.02
14	0.03
15	0.00



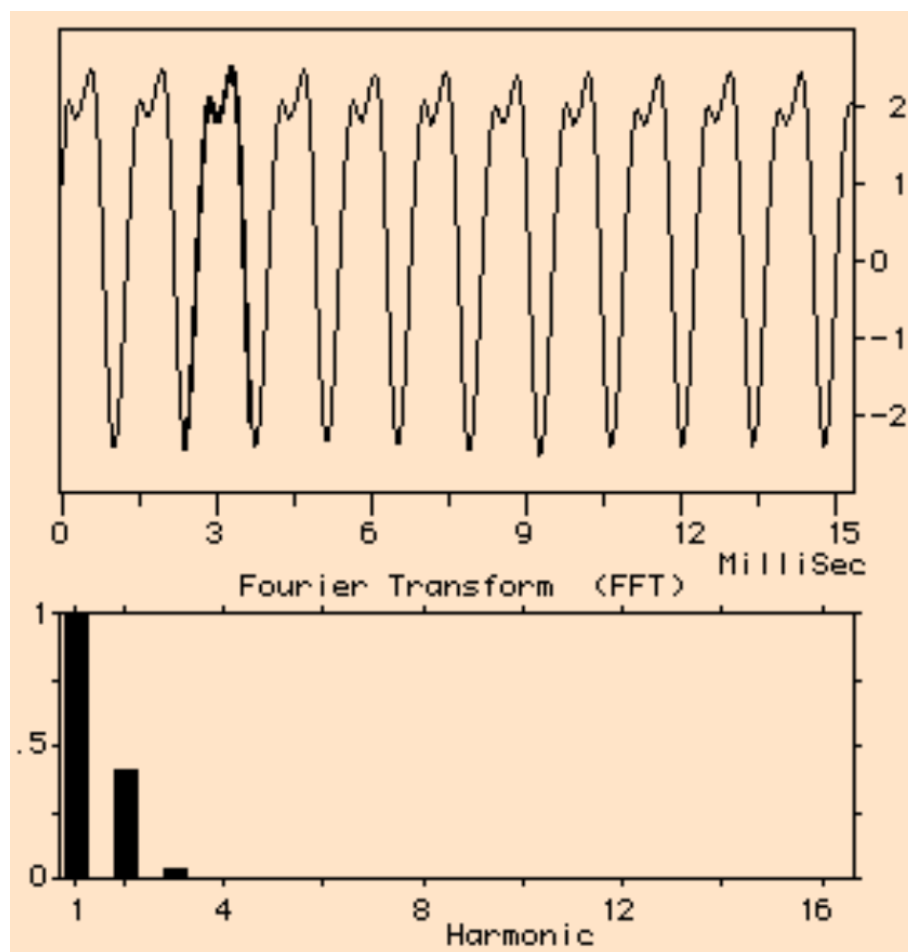
- Nota Bb3 a 233 Hz

Harmònic	Amplitud
1	1.00
2	0.08
3	0.82
4	0.07
5	0.29
6	0.35
7	0.06
8	0.10
9	0.03
10	0.01
11	0.02
12	0.02
13	0.01
14	0.00



- Nota F5 a 698 Hz

Harmònic	Amplitud
1	1
2	0.42
3	0.04
4	0.01
5	0.00
6	0.01
7	0.00

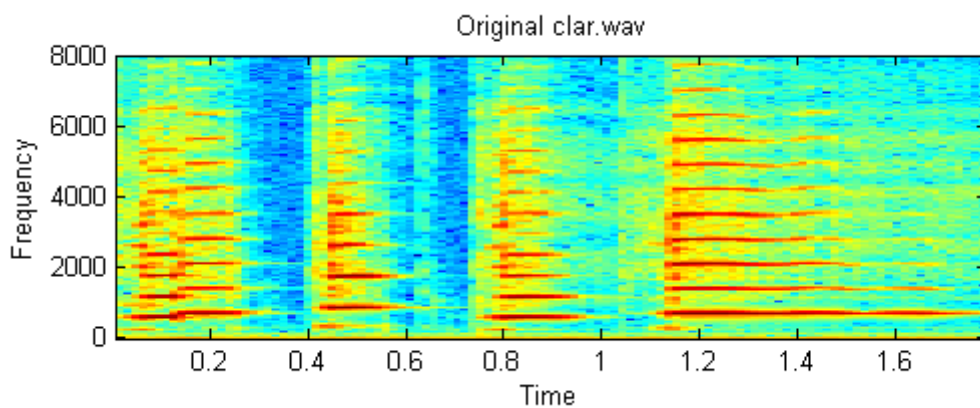


Annex F – Script MatLab Utilitzant Model d'Anàlisi/Síntesi Sinusoïdal + Soroll

Anàlisi sinusoïdal

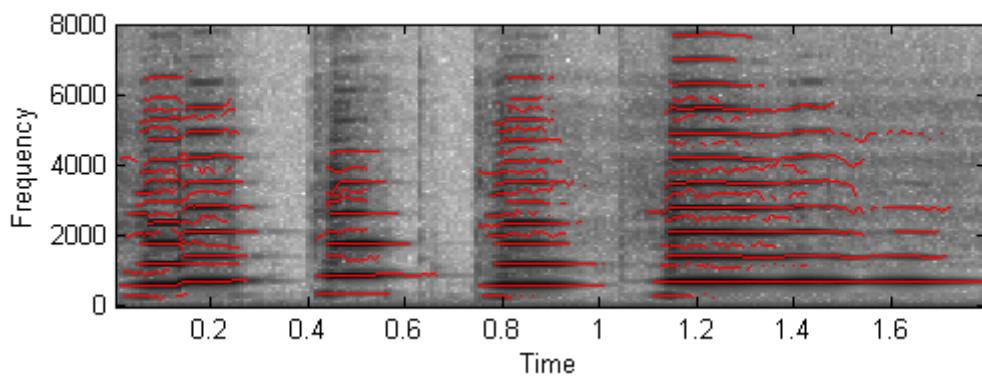
Realitzant la transformada de Fourier discreta en el temps se n'obté l'espectrograma per extreure'n les magnituds espectrals al llarg del temps.

```
[d,sr] = wavread('clar.wav');  
subplot(211)  
specgram(d,512,sr);  
title('Original clar.wav');
```



Amb la funció $[R,M]=\langle extractrax.m \ extractrax \rangle(S,T)$ es du a terme l'extracció de pics per quedar-se amb els que tenen més energia. D'aquesta manera s'obté la part harmònica de la senyal que són les línies de color vermell del següent gràfic:

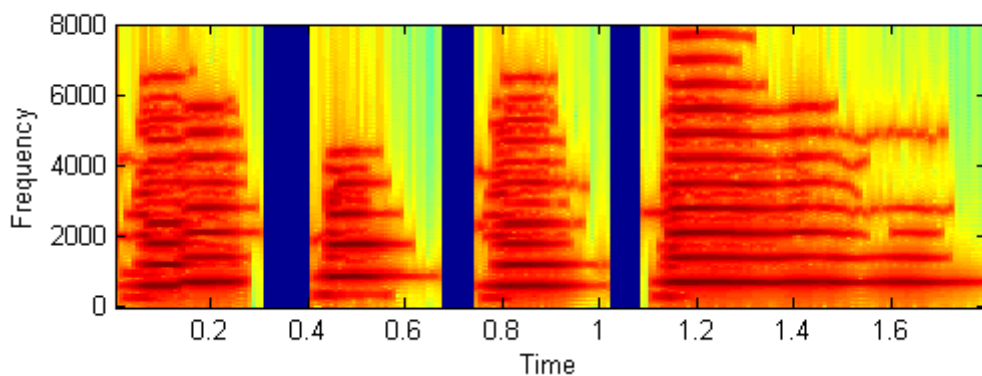
```
[d,sr] = wavread('clar.wav');  
S = specgram(d,256); % SP Toolbox routine (or use ifgram.m below)  
[R,M]=extractrax(abs(S)); % find peaks in STFT *magnitude*  
disp(['size of R is ', num2str(size(R,1)), ' rows x ', num2str(size(R,2)), ' cols']);  
tt = [1:224]*128/sr; % default specgram step is NFFT/2 i.e. 128  
F = R*sr/256; % Convert R from bins to Hz  
specgram(d,256,sr)  
colormap(1-gray) % black is intense, white is quiet  
hold on  
plot(tt,F,'r'); % the tracks follow the specgram peaks
```

Re-Síntesi

A continuació es re-sintetitza el senyal amb els harmònics extrets anteriorment a partir de la funció $X = \text{synthtrax}(F, M, SR, W, H)$. Es a dir es crea una senyal nova a partir dels sinus harmònics extrets de la original, deixant de banda la part inharmònica i/o soroll.

```
dr1 = synthtrax(F, M/64, sr, 256, 128); % Divide M by 64 to factor out window, FFT weighting
specgram(dr1, 256, sr)
sound(dr1, sr)
```



Extracció residual

A partir del senyal original i el que s'ha re-sintetitzat amb només la part harmònica, ara es fa un procés d'obtenció de la part residual (inharmònic + soroll), que està composta per sorolls de la gravació, de les respiracions o el mateix soroll de l'aire que s'està bufant a l'instrument.

Com que en aquest procés es necessita cancel·lar els harmònics, la fase pren molta importància, i per això s'utilitzaran funcions noves com ara:

$[I, S] = \text{ifgram}(X, N, W, H, SR)$ calculates both a conventional spectrogram (retuned in S) and an 'instantaneous frequency' gram, formed by taking the discrete derivative of the phase in each STFT frequency channel along time. This permits a more accurate estimate of peak frequencies. X is the sound, N is the FFT length, W is the time window length, H is the hop advance, and SR is the sampling rate (to get the frequencies scaled right).

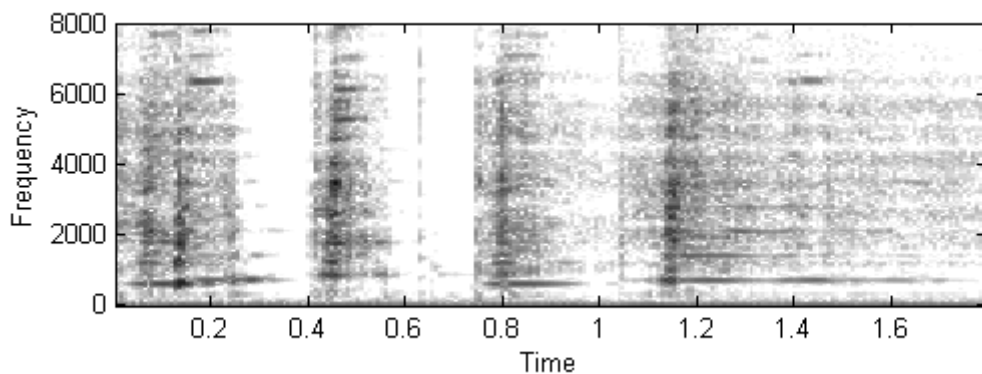
$V = \text{colinterpvals}(R,M)$ interpolates values down the columns of a matrix. R is a set of fractional indices (starting from 1.0, possibly including NaNs for missing points); V is returned as a conformal matrix, with each value the linear interpolation between the bins of the corresponding column of matrix M .

$X = \text{synthphtrax}(F,M,P,SR,W,H)$ performs sinusoid resynthesis just like `synthtrax`, but this version takes a matrix of exact phase values, P , to which the oscillators must conform at each control time. Matching both frequencies (from F) and phases (from P) requires cubic phase interpolation, since frequency is the time-derivative of phase.

Seguidament es mostra el gràfic d'aquesta part residual, on s'aprecia en comparació a les anteriors la poca energia espectral que representa.

```
% Calculate spectrogram and inst. freq. -gram
% with 256 pt FFT & window, 128 pt advance
[I, S]=ifgram(d, 256, 256, 128, sr);
% Extract the peak tracks based on the new STFT
[R, M]=extractrax(abs(S));
% Calculate the interpolated IF-gram values for exact track frequencies
F = colinterpvals(R, I);
% Interpolate the (columnwise unwrapped) STFT phases to get exact peak
% phases for every sample point (synthesis phase is negative of analysis)
P = -colinterpvals(R, unwrap(angle(S)));
disp(['size of F is ', num2str(size(F, 1)), ' x ', num2str(size(F, 2))]);
fcols = size(F, 2);
% Pad each control matrix with an extra column at each end,
% to account for the N-H = H points lost fitting the first & last windows
F = [0*F(:, 1), F, 0*F(:, end)];
M = [0*M(:, 1), M, 0*M(:, end)];
P = [0*P(:, 1), P, 0*P(:, end)];
% (multiplying nearest column by 0 preserves pattern of NaN values)

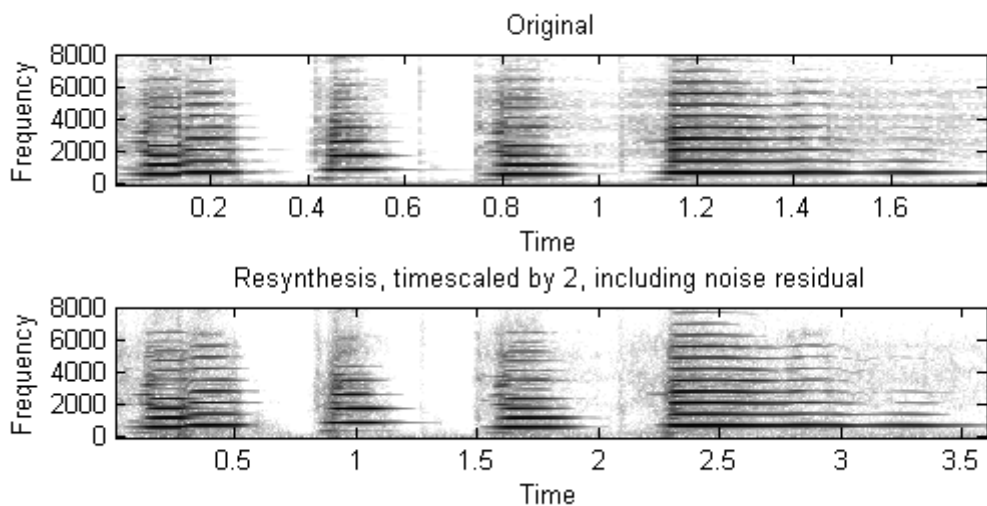
% Now, the phase preserving resynthesis:
dr2 = synthphtrax(F, M, P, sr, 256, 128);
sound(dr2, sr)
% Noise residual is original with harmonic reconstruction subtracted off
dre = d - dr2(1:length(d))';
specgram(dre, 256, sr);
colormap(1-gray)
axis([-60 0])
sound(dre, sr)
```



Modificant la re-Síntesi

Un cop es té el senyal separat entre la part harmònica i la de soroll ja s'hi pot treballar modificant-les per separat, ja que cada una de les parts necessita tècniques diferents per després poder fer una re-síntesi tal i com es desitgi. En aquest cas es procedeix a fer una alteració en quant a la durada del senyal, es a dir, passarà a durar el doble, però s'aconseguirà mitjançant una re-síntesi que permet mantenir freqüències i comportaments harmònics.

```
% LPC model of residual
[a,g,e] = lpcfit(dre, 12, 256, 64);
% Use hopsize 64 = N/4 so it will still sound OK time expanded
dre2 = lpcsynth(a, g, [], 256, 128); % doubled hop of 128 time expands
% Use (non-phase) synthtrax since we don't care about phase
% hopsize = 256 is 2x expanded compared to analysis hopsize=128
drs2 = synthtrax(F,M,sr,512,256);
% Add them together, making sure to match sample counts
mi_nlen = min(length(drs2),length(dre2));
dx2 = drs2(1:mi_nlen)+dre2(1:mi_nlen);
% Compare spectrogram to original
subplot(311)
specgram(d,256,sr)
title('Original');
axis([-60 10])
subplot(312)
specgram(dx2,256,sr)
axis([-60 10])
title('Resynthesis, timescaled by 2, including noise residual')
colormap(1-gray)
sound(dx2,sr)
```



Aquesta última figura demostra com la senyal original i la re-escalada en el temps, són pràcticament idèntiques exceptuant que la línia temporal és del doble a la re-sintetitzada.